

Programming Boot Camp

# Database design, Data manipulation, External integration with Adalo

Tokyo Institute of Technology 2022/11/13

Ryo Imahashi

# Table of Contents

- Review previous lecture and check the goal of this lecture
- Introduction to databases
- Database design
- Data manipulation
- Improve sample application
- External integration
- Team development with Adalo
- Exercises
- Summary

## Review previous lecture and check the goal of this lecture

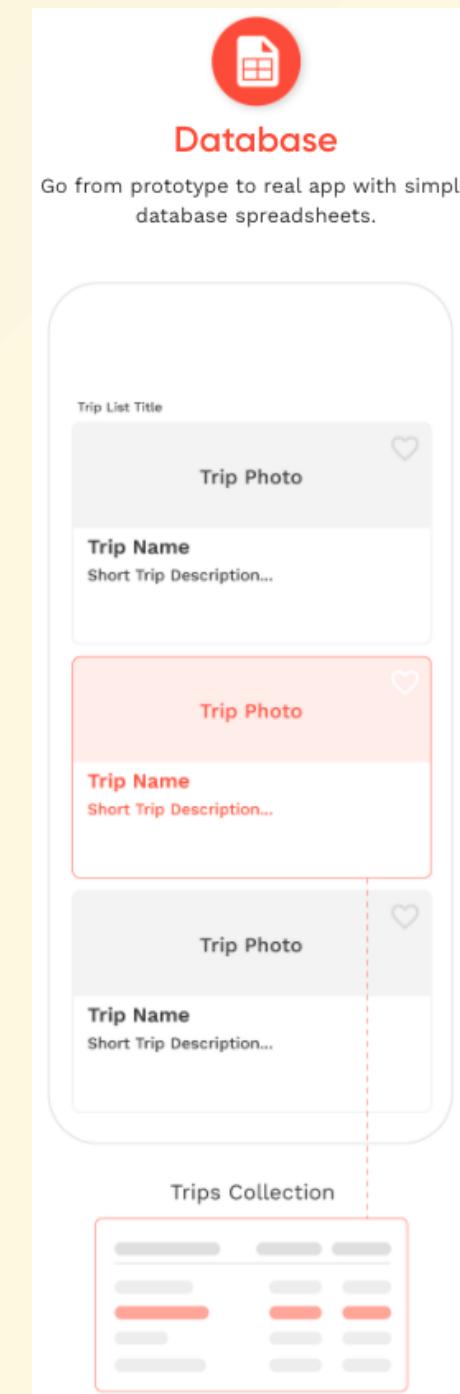
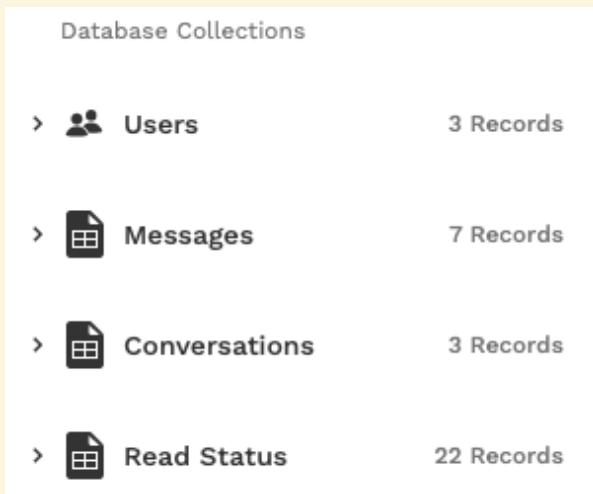
- In the previous lecture, we learned Adalo, a no-code tool, and created the UI of the pet health management app.
  - In the lecture, we used simple components that does not require a database.
- In this lecture, we will design a database that matches the UI we created in the previous lecture, so that we can manipulate the data from the app.
- After that, we will improve the sample app learning some features of Adalo, learn external integration and team development. Finally, we will do app development exercises and presentation.

# Introduction to databases

First, let's learn about databases.

# Database(Previous lesson)

- A set of organized data.
- Data can be registered, read (displayed), updated, and deleted.
- Example: In the case of a Chat application



- Databases are often compared to "spreadsheet-like" software.
- A database can be used to CREATE, READ, UPDATE, and DELETE data. These operations are collectively called CRUD operations.

Users						<a href="#">+ ADD USER</a>				
	<a href="#">A</a> Email	Password	<a href="#">A</a> Username	<a href="#">A</a> Full Name	Profile Picture					
<input type="checkbox"/>	fuga@hoge.com	[hidden]		三人目のユーザー						
<input type="checkbox"/>	hoge@fuga.com	[hidden]		東 工大						
<input type="checkbox"/>	imahashi@example.com	[hidden]		今橋 謙						

# Basics of Adalo's database



You can access Adalo's database from this icon.  
There are three components of the Adalo database.

- Collection
- Property
- Record

# What is a Collection?

A collection of data that has the same property.

The screenshot shows a database interface with a sidebar on the left and a main table view on the right.

**Left Sidebar:**

- + Database Collections
- Database Collections
- Users (3 Records)
- ⋮
- Email
- Password
- Username
- Full Name
- Profile Picture
- Conversations (Creator)s
- Messages (Sender)s
- Read Statuses
- Conversations (Members)s
- Friends
- + ADD PROPERTY
- Messages (7 Records)
- Conversations (3 Records)

**Main View:**

### Users

	A Email	>Password	A Username	A Full Name	Profile Picture
<input type="checkbox"/>	fuga@hoge.com	[hidden]		三人目のユーザー	
<input type="checkbox"/>	hoge@fuga.com	[hidden]		東 工大	
<input type="checkbox"/>	imahashi@example.com	[hidden]		今橋 陵	

**Top Right:** + ADD USER, upload, download, sync, search.

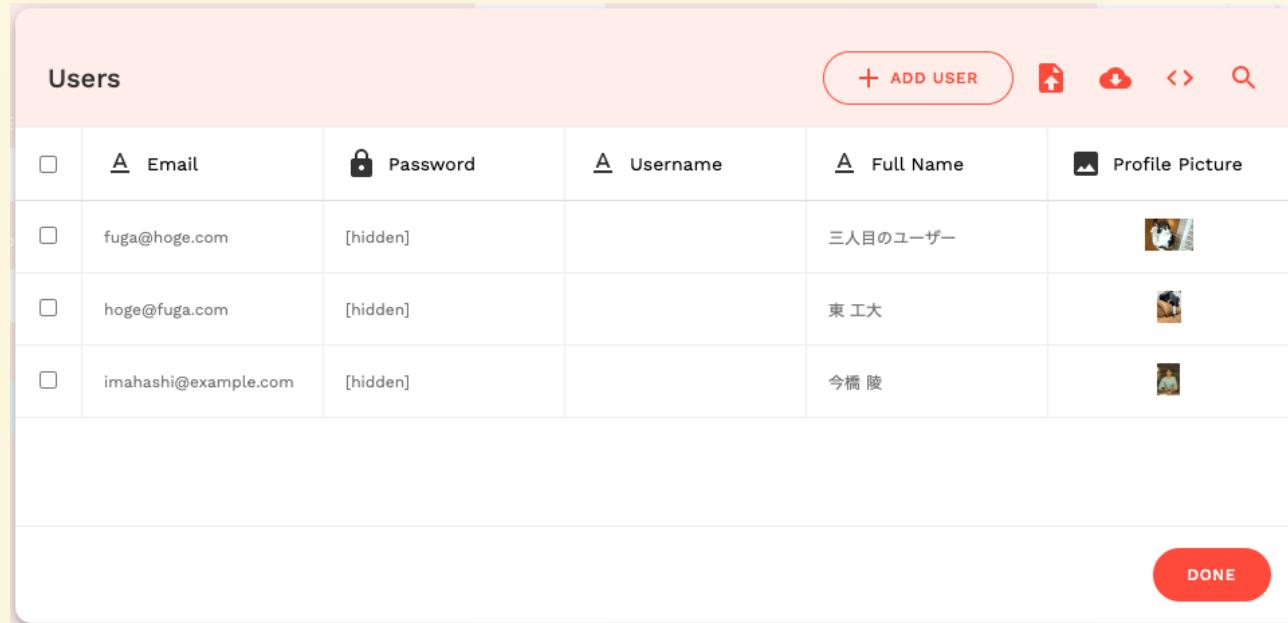
**Bottom Right:** DONE

- Collection is used to divide and organize the various data.(An analogous term is table.)
- In many cases, a Collection is a group of data that a user can register, update, or delete in a single operation. <!-- (A collection is often said to be something that can be expressed as a noun.
- By default, Users is prepared as a Collection, and the rest can be added according to the application to be developed.

It is very difficult to decide what kind of collection to add. Let's practice and get used to it. (If you have any problems, consult with mentors.)

## What is Record?

- Record is a unit of information in a collection.
  - One row corresponds to one Record.
- In the example of Users Collection, the information belongs to one user is registered as one Record.



The screenshot shows a MongoDB Compass interface with a 'Users' collection. The collection contains four documents, each represented by a row in a table. The columns are labeled: Email, Password, Username, Full Name, and Profile Picture. The first document has an Email of 'fuga@hoge.com' and a Username of '三人目のユーザー'. The second document has an Email of 'hoge@fuga.com' and a Username of '東 工大'. The third document has an Email of 'imahashi@example.com' and a Username of '今橋 賢'. Each document also has a checkbox in the first column and a small preview image of the profile picture in the fifth column. At the top right of the table header, there is a red button labeled '+ ADD USER' and other icons for file operations and search.

	Email	Password	Username	Full Name	Profile Picture
<input type="checkbox"/>	fuga@hoge.com	[hidden]	三人目のユーザー		
<input type="checkbox"/>	hoge@fuga.com	[hidden]		東 工大	
<input type="checkbox"/>	imahashi@example.com	[hidden]		今橋 賢	

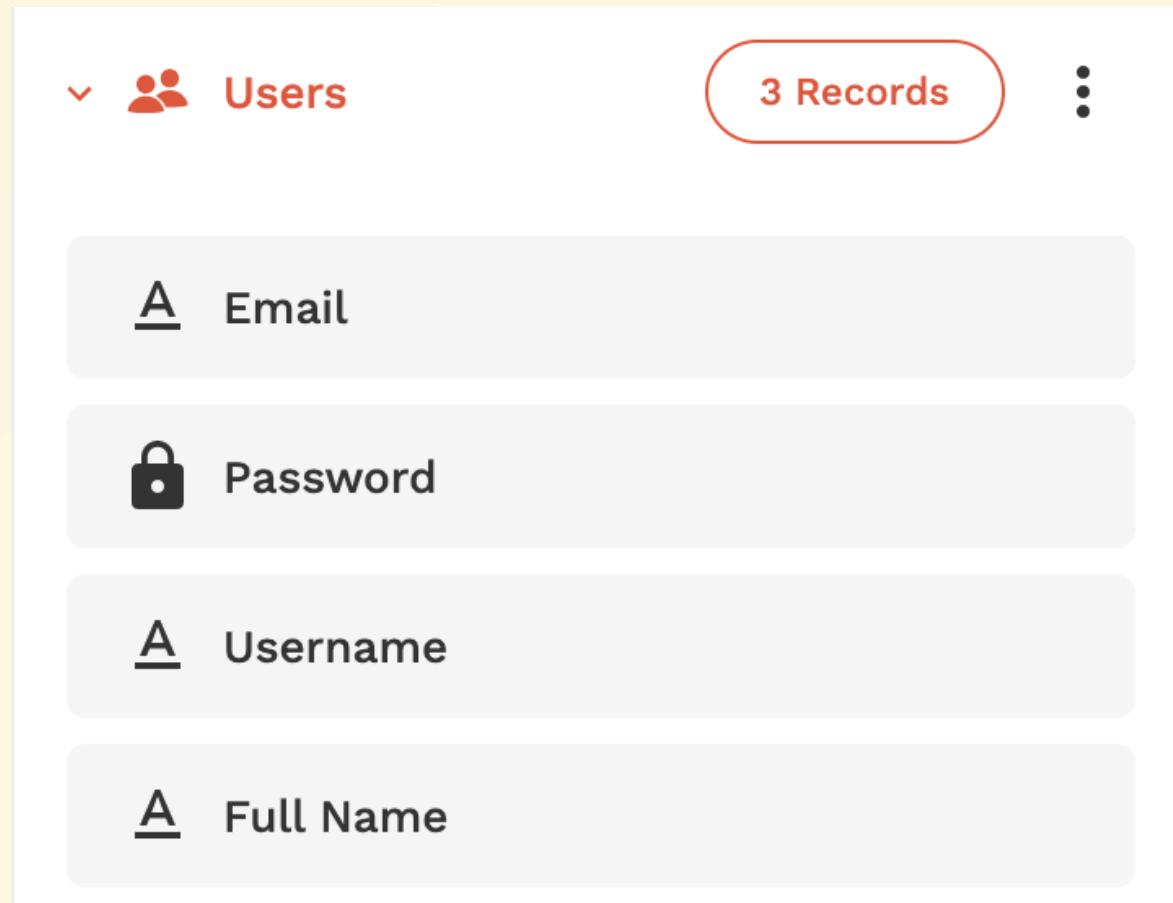
DONE

- Records are basically registered from the form on the screen of the app, but it is also possible to register from the form by pressing the "+Add xxxx" button on the upper right in Record View.
- You can also search for Records in the Collection, and upload (import) and download CSV files.

The screenshot shows a mobile application interface for creating a new user account. The title bar at the top left says "New User". The form consists of five input fields: "Email" (with an "A" icon), "Password" (with a lock icon), "Username" (with an "A" icon), "Full Name" (with an "A" icon), and "Profile Picture" (with a camera icon). Below the profile picture field is a placeholder text "Choose image". At the bottom right are two buttons: "CANCEL" and a red "SAVE" button.

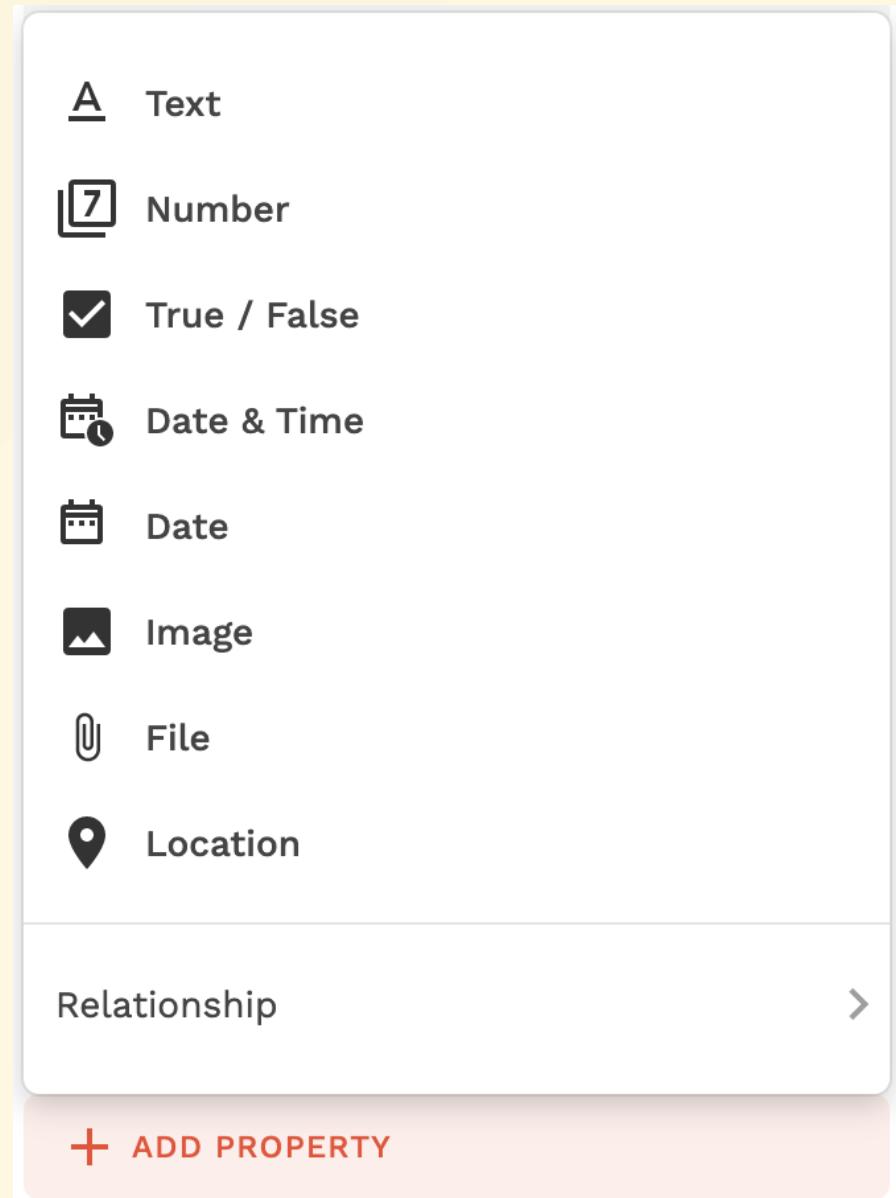
## What is Property?

- Property is each and every item that makes up a Record.
- The Users Collection consists of properties such as email, password, user name, and name.
- The value of Property can be empty.



To define what kind of data the Property is, select the Type when adding it.

- Text
- Number
- True/False
- Date/Time
- Date/Time
- Image
- File
- Location

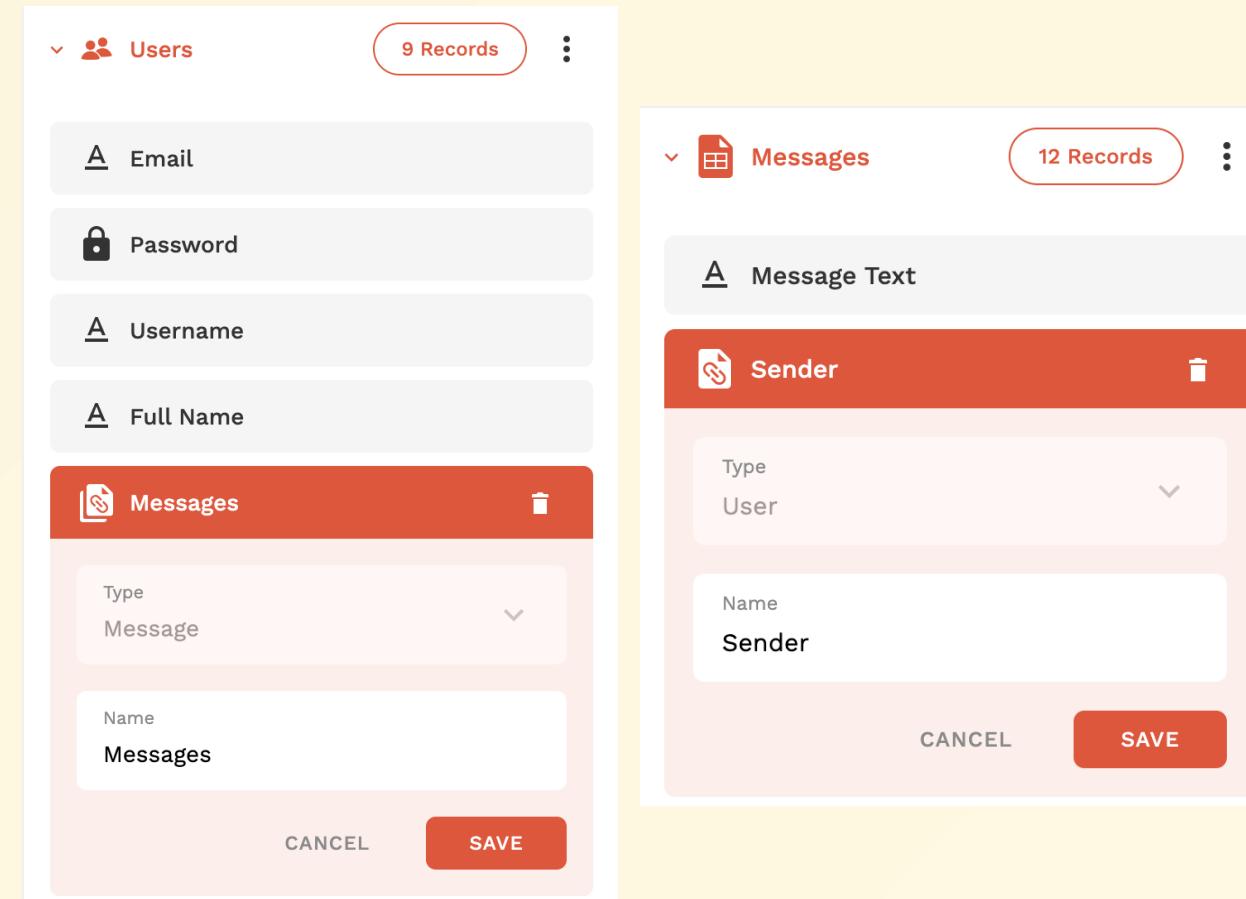


## What is Relationship?

- Instead of storing a large number of properties for a single Record, we can set a special property to relate multiple Collections, called Relationship. This allows you to divide a Collection into manageable pieces.

For example, a message sent by a user in the Chat app is stored in the Messages Collection, which is separate from the Users Collection, and these two collections are related by Relationship.

The Users side has a Relationship called Messages, and the Messages side has a Relationship called Sender (with Users).

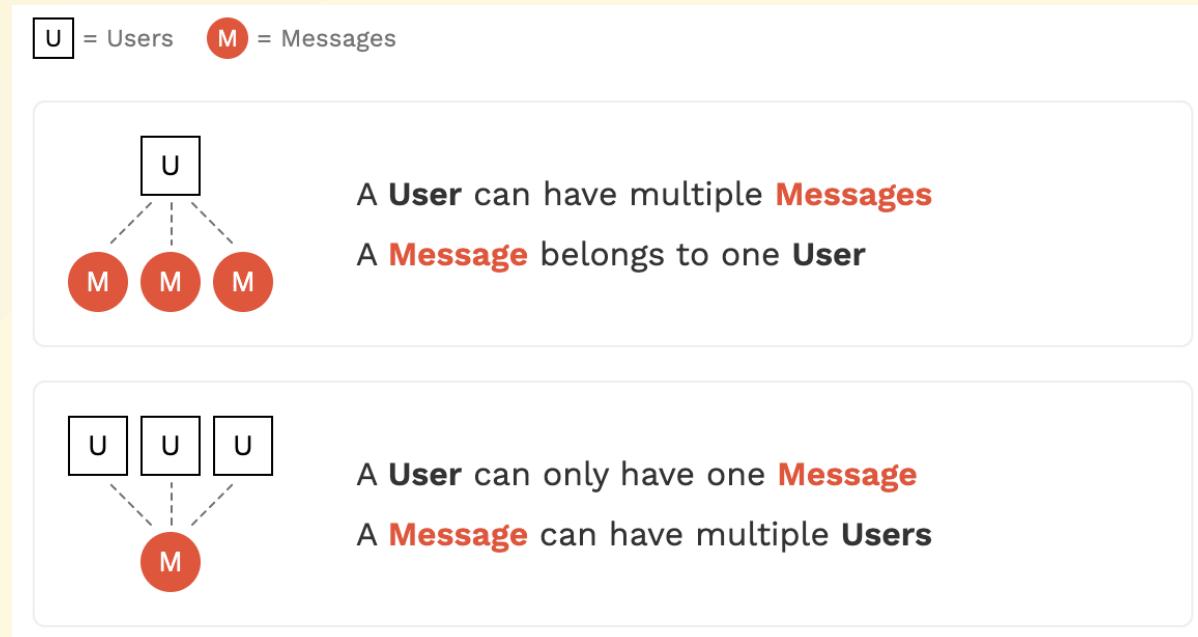


## Types of Relationship

- In Adalo's Relationship, you can choose one of two types, one-to-many or many-to-many, depending on the number of Records associated with the Collection.

# One-to-many Relationship

- This means that one Record has a relationship with multiple Records in different collections.
- Depending on whether the Collection you are trying to set the Relationship to is one or many, two choices will appear.



# Example of a one-to-many Relationship

- In the Chat application, one user sends multiple messages, but the sender of the message is one user, so the Relationship in the Users Collection and Messages Collection is one-to-many.

**Users**

	Email	Password	Full Name	Conversations	Messages
<input type="checkbox"/>	kyogoku@guildworks.jp	[hidden]	naotake	1 conversation	1 message
<input type="checkbox"/>	ueno@guildworks.jp	[hidden]	jun116	1 conversation	2 messages
<input type="checkbox"/>	imahashi@guildworks.jp	[hidden]	Ryo Imahashi	4 conversations	9 messages
<input type="checkbox"/>	mike@example.com	[hidden]	Michael Scott	1 conversation	
<input type="checkbox"/>	pam@example.com	[hidden]	Pam Beesly		
<input type="checkbox"/>	elon@spacex.com	[hidden]	Elon Musk	1 conversation	

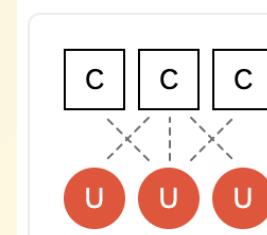
  

**Messages**

	Message Text	Sender	Conversation	Created	Updated
<input type="checkbox"/>	Hi!	imahashi@guildworks.jp	naotake / Ryo Imahashi	5 months ago	5 months ago
<input type="checkbox"/>	Hi 🌟	kyogoku@guildworks.jp	naotake / Ryo Imahashi	5 months ago	5 months ago
<input type="checkbox"/>	(ΦωΦ)777...	ueno@guildworks.jp	Ryo Imahashi / jun116	5 months ago	5 months ago
<input type="checkbox"/>	タ――――( ∀ )――――!!	ueno@guildworks.jp	Ryo Imahashi / jun116	5 months ago	5 months ago
<input type="checkbox"/>	hi	imahashi@guildworks.jp	Ryo Imahashi / jun116	5 months ago	5 months ago
<input type="checkbox"/>	Hi	imahashi@guildworks.jp	Ryo Imahashi / Michael...	5 months ago	5 months ago
<input type="checkbox"/>	こんにちは！	imahashi@guildworks.jp	Ryo Imahashi / Michael...	5 months ago	5 months ago

# Many-to-many Relationships

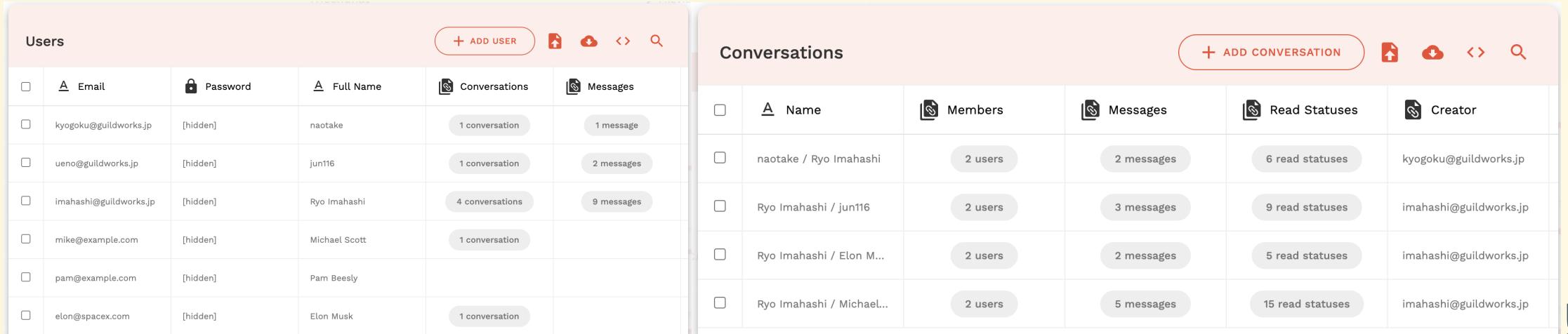
- This means that one Record in both Collections is tied to multiple Records in the other Collection.



A **User** can have multiple **Conversations**  
A **Conversation** can have multiple **Users**

# An example of a many-to-many Relationship

- In a Chat application, one user can have multiple conversations (to keep track of who and what messages were exchanged), and one conversation can have multiple members (users), so the relationship between the Users Collection and the Conversations Collection is many-to-many.



The screenshot displays two tables representing collections in a database:

Users					
	<input type="text"/> Email	<input type="password"/> Password	<input type="text"/> Full Name	<input type="button"/> Conversations	<input type="button"/> Messages
<input type="checkbox"/>	kyogoku@guildworks.jp	[hidden]	naotake	1 conversation	1 message
<input type="checkbox"/>	ueno@guildworks.jp	[hidden]	jun116	1 conversation	2 messages
<input type="checkbox"/>	imahashi@guildworks.jp	[hidden]	Ryo Imahashi	4 conversations	9 messages
<input type="checkbox"/>	mike@example.com	[hidden]	Michael Scott	1 conversation	
<input type="checkbox"/>	pam@example.com	[hidden]	Pam Beesly		
<input type="checkbox"/>	elon@spacex.com	[hidden]	Elon Musk	1 conversation	

Conversations					
	<input type="text"/> Name	<input type="button"/> Members	<input type="button"/> Messages	<input type="button"/> Read Statuses	<input type="button"/> Creator
<input type="checkbox"/>	naotake / Ryo Imahashi	2 users	2 messages	6 read statuses	kyogoku@guildworks.jp
<input type="checkbox"/>	Ryo Imahashi / jun116	2 users	3 messages	9 read statuses	imahashi@guildworks.jp
<input type="checkbox"/>	Ryo Imahashi / Elon M...	2 users	2 messages	5 read statuses	imahashi@guildworks.jp
<input type="checkbox"/>	Ryo Imahashi / Michael...	2 users	5 messages	15 read statuses	imahashi@guildworks.jp

!

# Database design

Let's design the database by looking at the UI of the sample app created in the previous lecture and thinking about what kind of data needs to be saved.

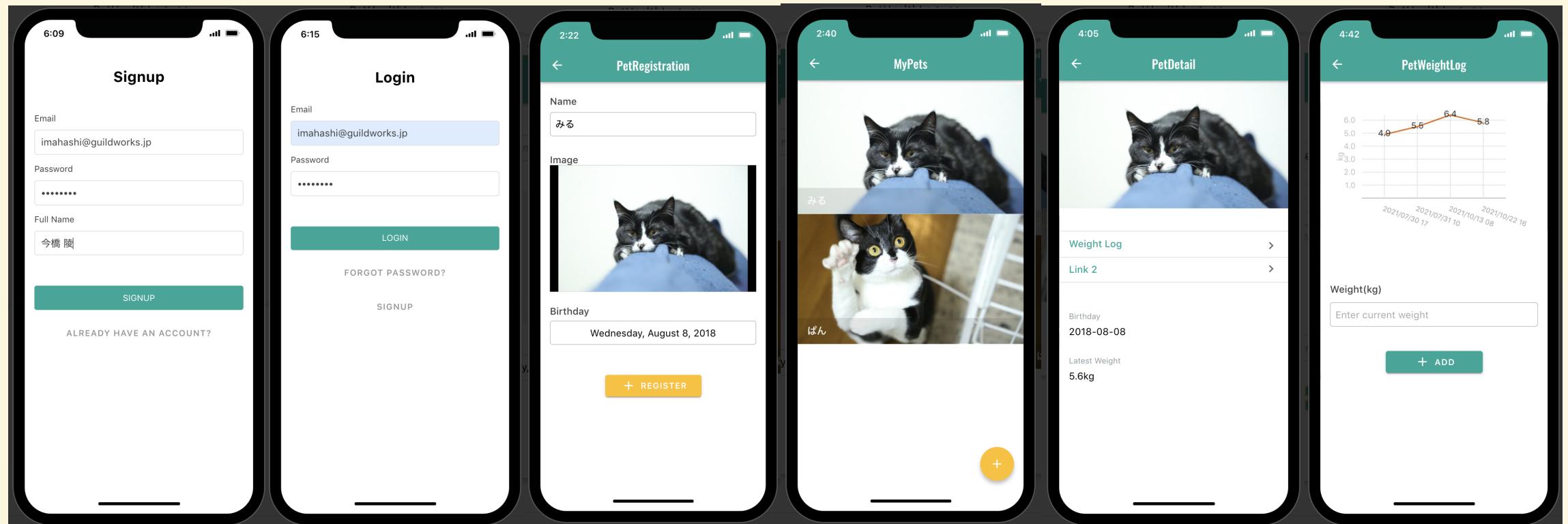
## URL for cloning the app created in the previous lecture

- Please clone the app from the following URL. We will use it to proceed with the lecture from here.

<https://previewer.adalo.com/014fd9d1-80c6-4325-899a-d943e778c865>

# Let's design the database.

Let's design the database by looking at the UI of the sample application. The steps are described in the next page.



## Database Design Steps

1. While looking at the UI, make a list of the data that will need to be saved. Write them down in a text editor (e.g. Notepad application).
2. Think about what kind of collections should be created to store the listed data, and create the collections in the Adalo database.
3. Add the data you listed in step 1 as a Property to the appropriate Collection and select the appropriate Type.
4. Set the Relationship Property to collections related to other collection.

In the next slide and onwards, there are explanations, but it is highly recommended that you try it by yourself before checking them.

There is no absolute right answer. When in doubt, follow your intuitions.

## Explanations

While looking at the UI, made a list of the data that needs to be saved, and it looked like this

- User's Email
- User's Password
- User's FullName
- Pet's Name
- Pet's Photo
- Pet's Birth Date
- Pet's weight
- Date and time the pet's weight was registered

- If anyone can name any other data, please let me know!

Thinking about what kind of collections should be created to store the listed data, listed these three Collections.

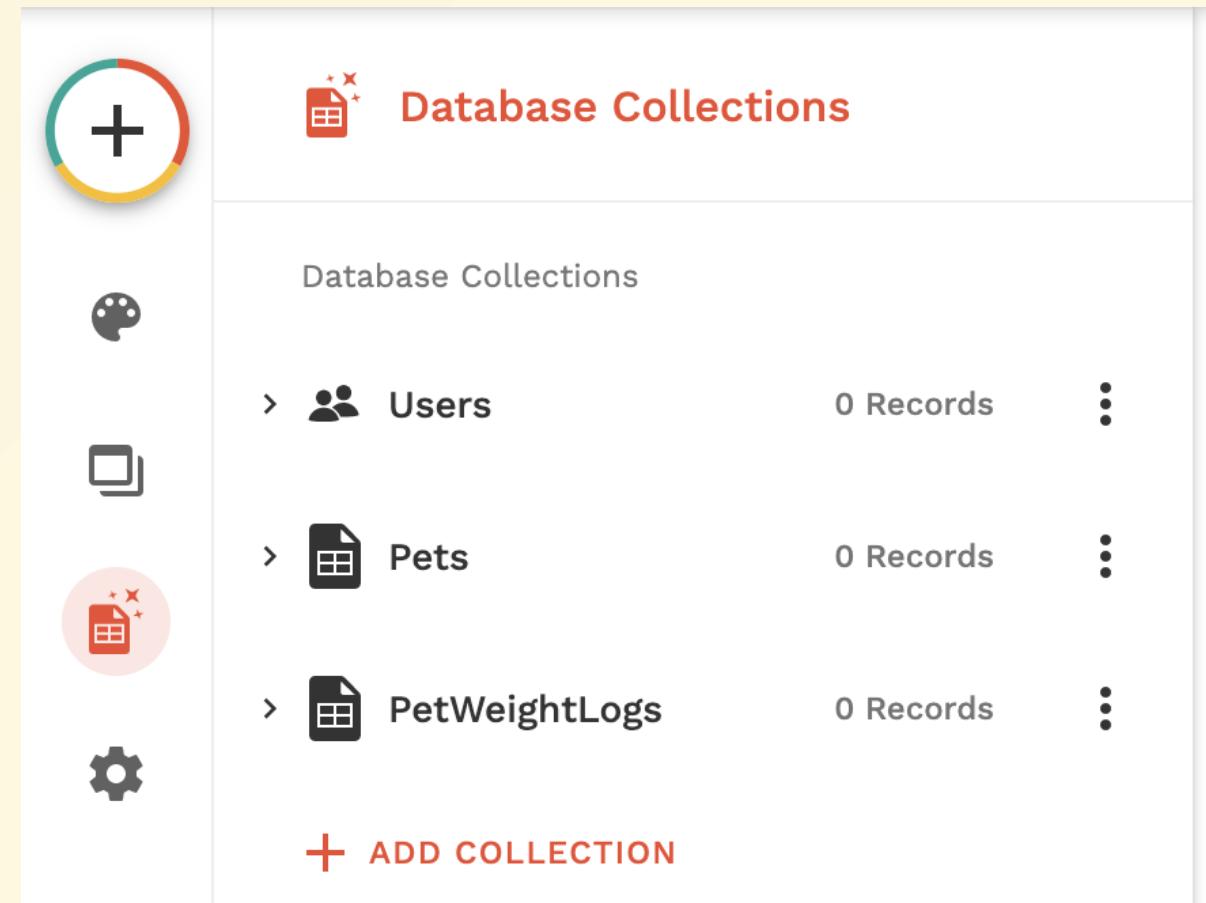
- Users
- Pets
- PetWeightLogs

- Many of you would have listed two collections, one for users and one for pets.
- Some people would not have listed a collection for pet weight records. (It is not wrong to include the pet's weight and its registration date in the pet's Collection. This will be explained later.)
- Have any of you listed other Collections?

## Additional information on Collection classification

- When the relationship "if A is identified, then B is identified" is true, "A" is often a Collection and "B" is a Property of that Collection.
  - If a user is identified, the user's email, password, and FullName are identified.
  - If a pet is identified, the pet's name, photo, and birthday are identified.
- When the relationship "there are multiple B's corresponding to A" is true, A and B are often split into two collections.
  - There are multiple pet weights and their registration times corresponding to a pet.

- Register the Collection into the Adalo database.
- Users is created by default.

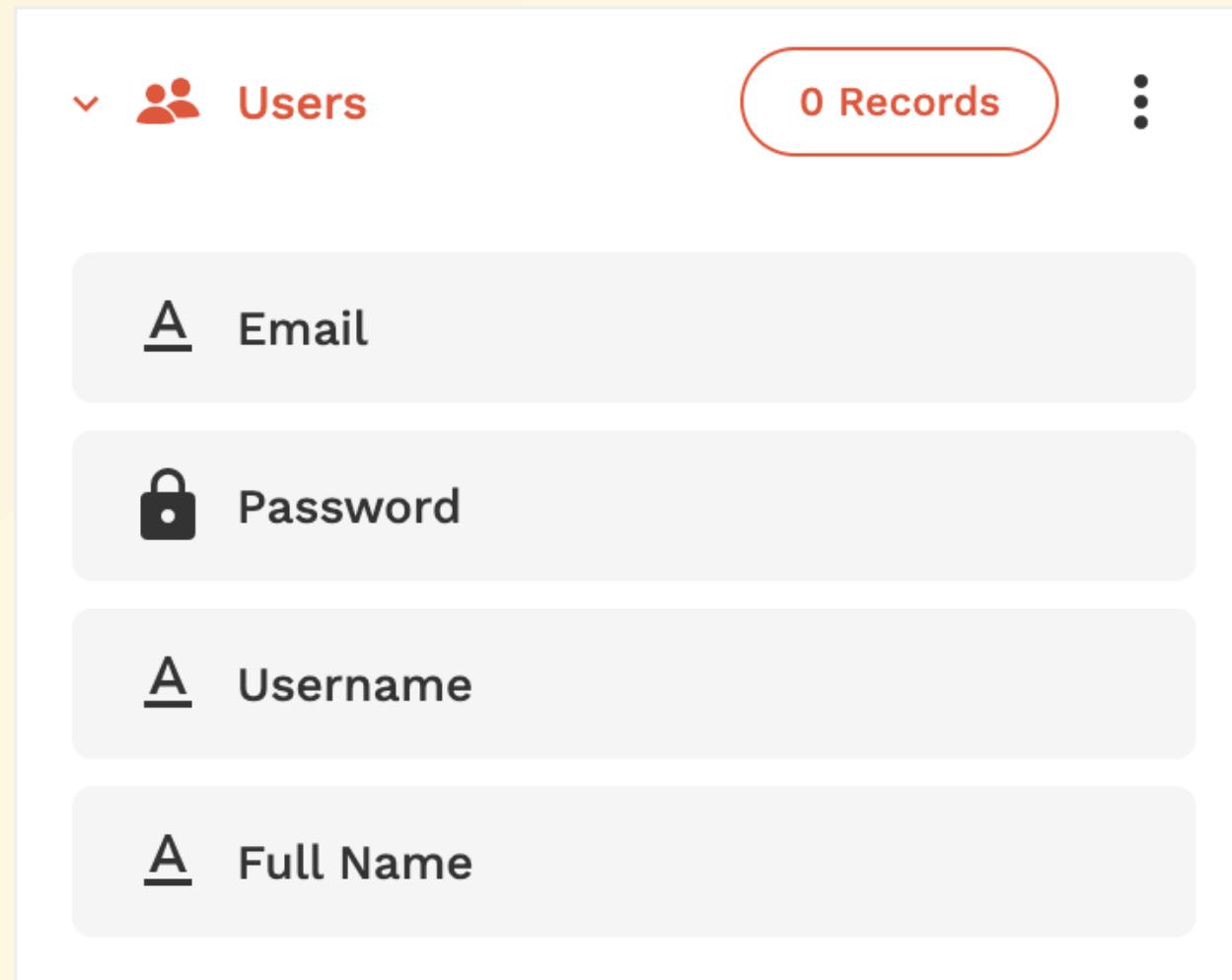


Next, I appended the data listed in 1 as a Property of the appropriate Collection. Types are placed in "()".

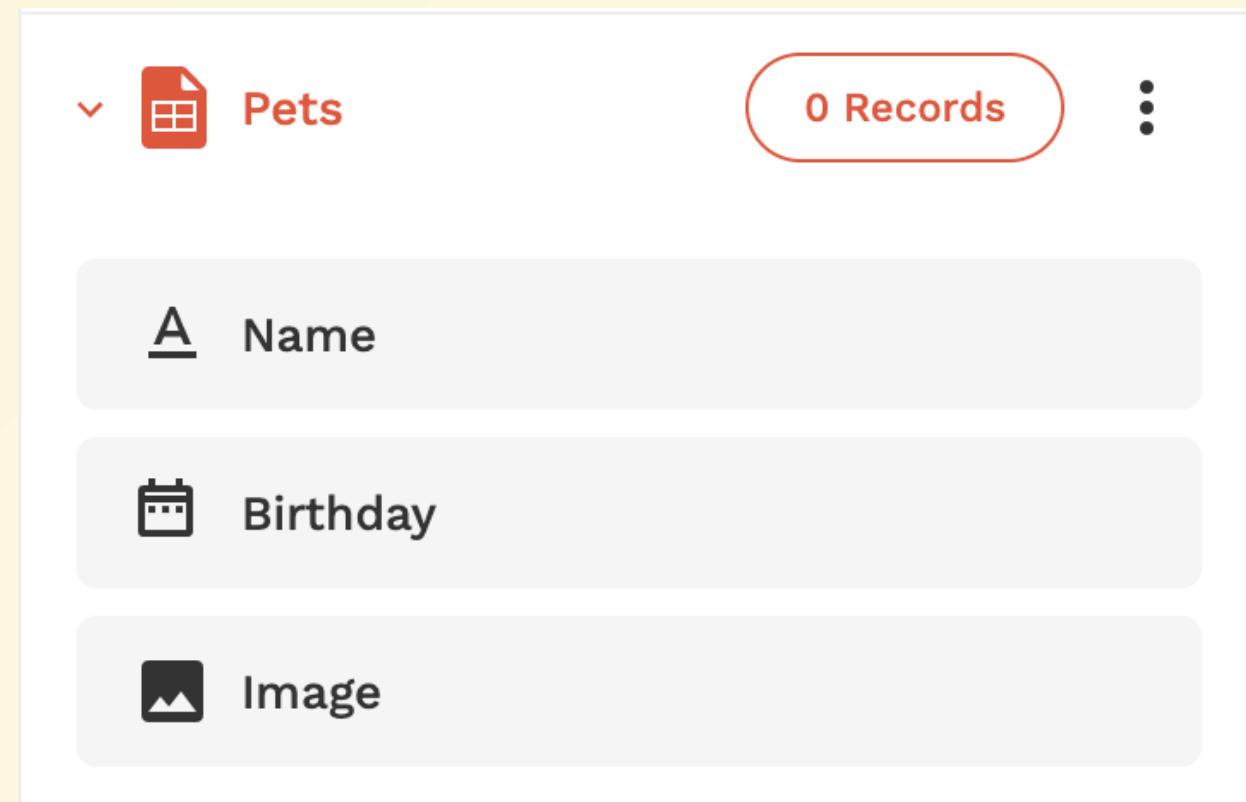
- Users
  - Email(Text)
  - Password(\*Password)
  - FullName(Text)
- Pets
  - Name(Text)
  - Image(Image)
  - Birthday(Date)
- PetWeightLogs
  - WeightKg(Number)
  - RegisteredTime(Date&Time)

Password is a special Type that is set to Password by default.

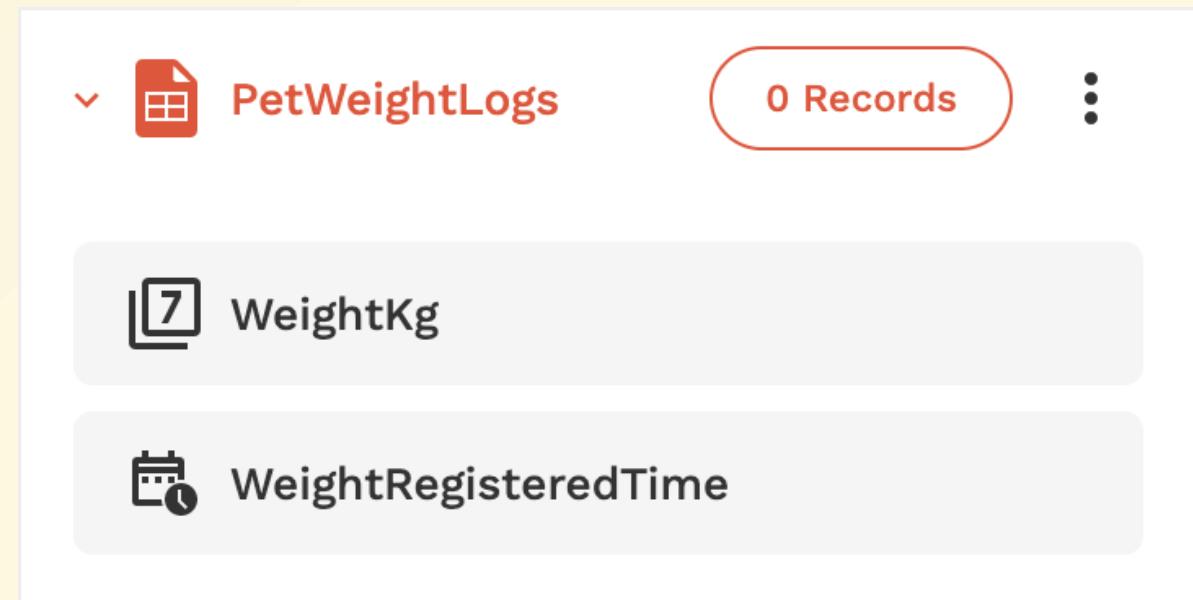
- In Adalo database, add the properties.
- The Users Collection is already set by default and contains all the necessary items.
- We don't need Username, but since we can't delete it, leave it as is.



- The Pets Collection Property looks like this.

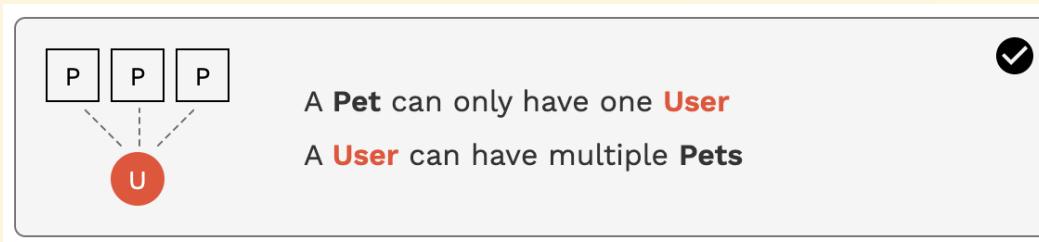


- The Property of the PetWeightLogs Collection looks like this.
- Delete the Name Property, which is set by default when you add a Collection.
  - You can delete it after dragging and dropping it so that the order is not at the top of the collection.



Finally, for collections that are related to other collections, set the Relationship Property.

- Select the Users Collection to add a one-to-many Relationship with the Pet Collection.



▼ **Users** 0 Records ⋮

**A Email**

**>Password**

**A Username**

**A Full Name**

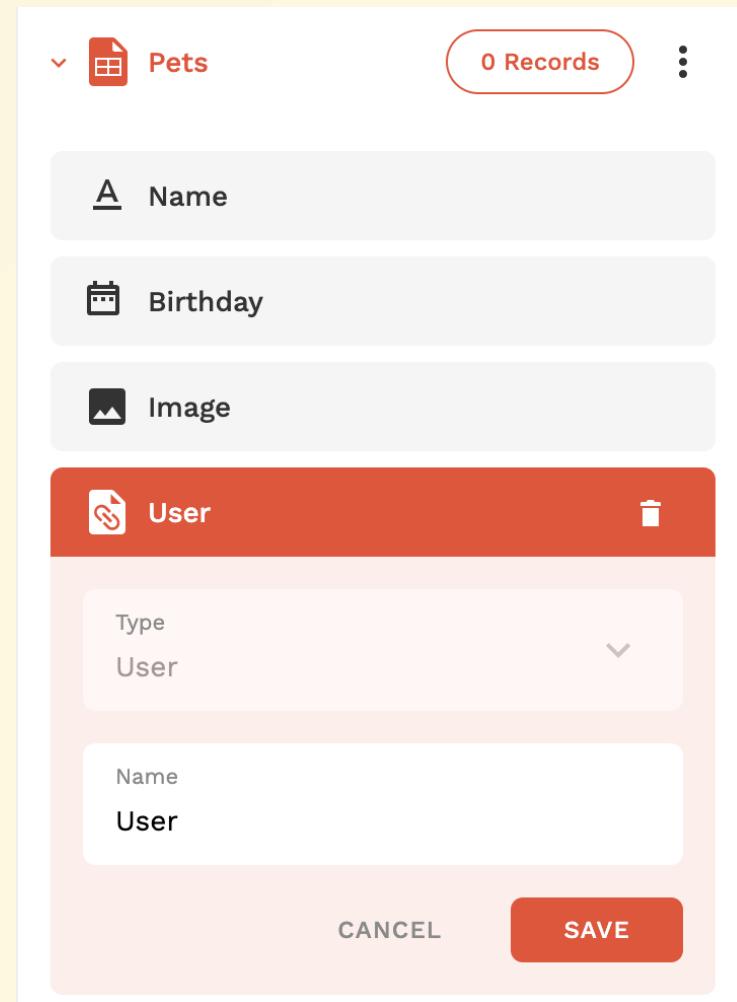
**Pets** trash icon

Type  
Pet

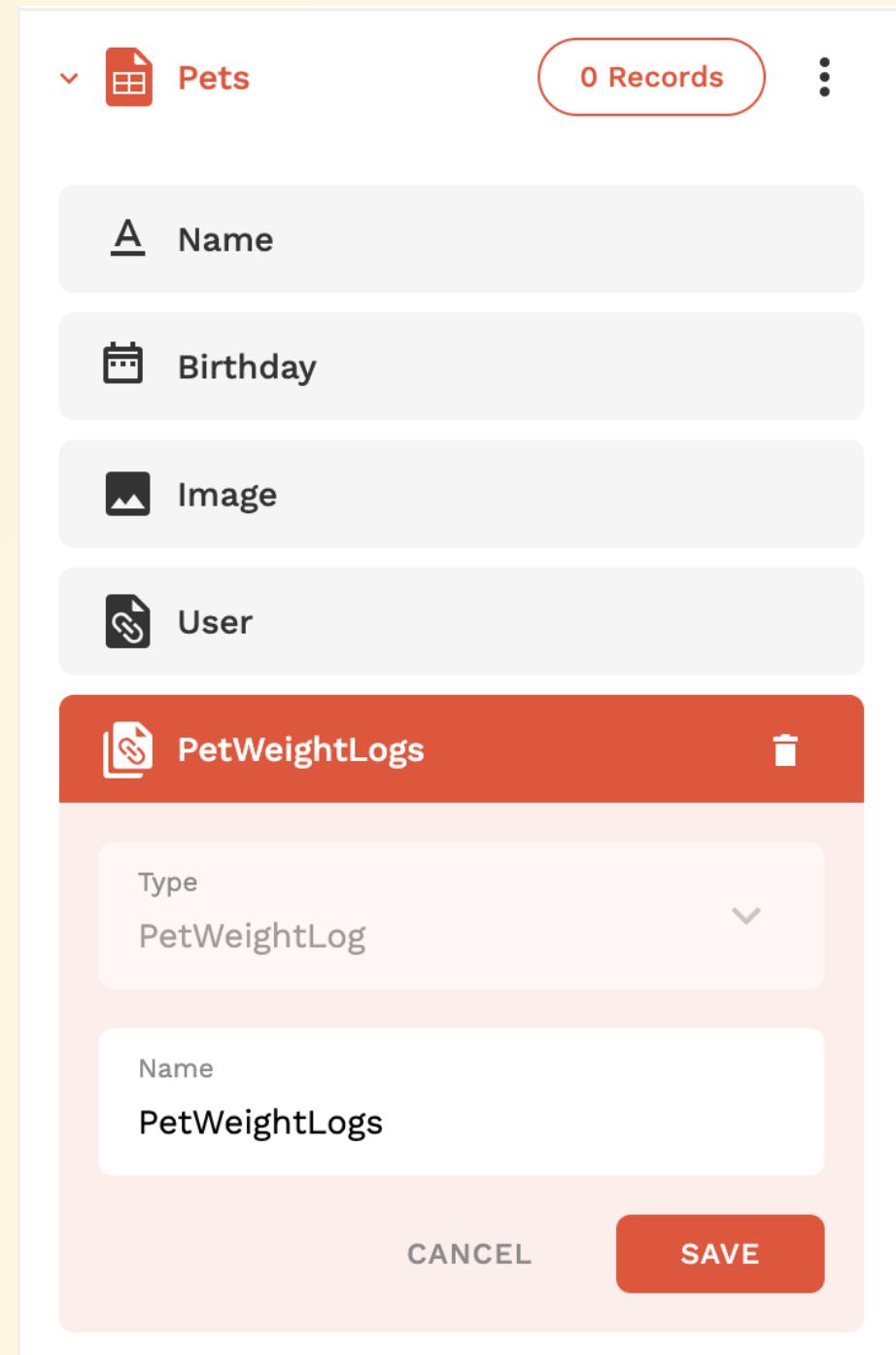
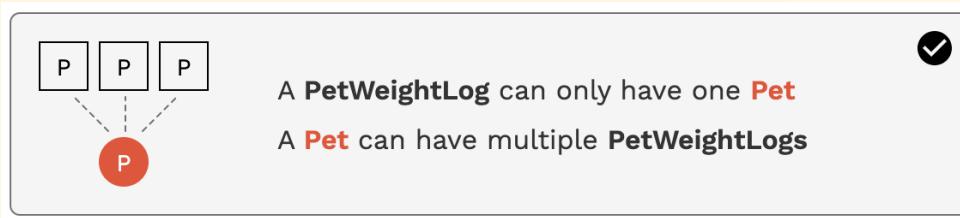
Name  
Pets

**CANCEL** **SAVE**

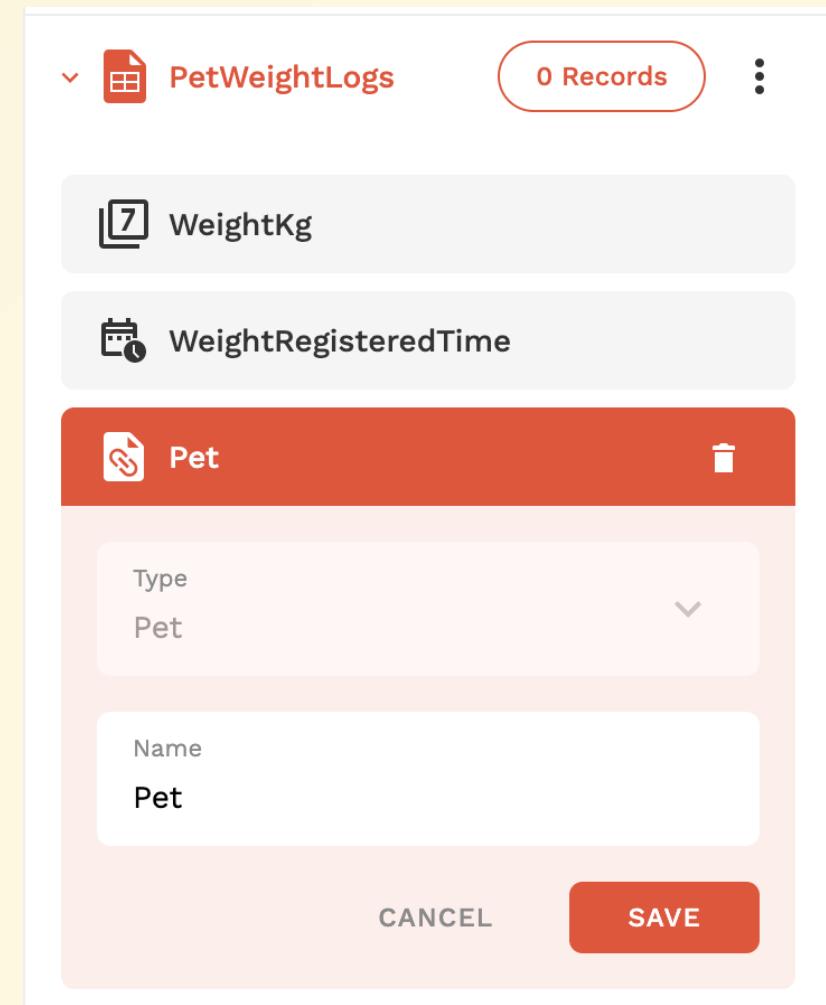
- If you check the Pets Collection, you will see that a Relationship with the Users Collection has been automatically added because the Relationship setting was made on the Users Collection side.
  - Since the Users Collection is one side, the "s" at the end is omitted and the Property name is "User".



- Add a Relationship to the Pets Collection with the PetWeightLogs Collection.
  - Select the Pets Collection to add a one-to-many Relationship with the PetWeightLogs Collection.



If you check the PetWeightLogs Collection, you will see that a Relationship with the Pets Collection has been automatically added because the Relationship setting was made on the Pets Collection side.



- Since the Pets Collection side is one side, the "s" at the end is omitted, and the Property name is "Pet".

# Reference: What happens if you include the pet weights and its registration times in the Pets Collection?

The record will be registered as follows, but in this case, you will have the troubles.

Pets						
	A Name	Image	Birthday	Weight(kg)	WeightRegisteredTime	Created
<input type="checkbox"/>	みる		8/8/2018	5.2	November 3, 2021 12:00 AM	a few seconds ago
<input type="checkbox"/>	みる		8/8/2018	5.1	November 2, 2021 12:00 AM	a minute ago
<input type="checkbox"/>	みる		8/8/2018	5	November 1, 2021 12:00 AM	a minute ago

## The troubles

- Because multiple Records with different pet weights and their registration times are registered for one pet, the pet's information (Name, Image, Birthday) is registered in duplicate.
  - In order to change the information of one pet, we have to update all the duplicated records, which makes the process more complicated.
- Adalo has a convenient function to automatically generate a form to register a record in a single collection, but since the collection is not divided by the unit to register data, it cannot be used.

That's it for the sample app database design.

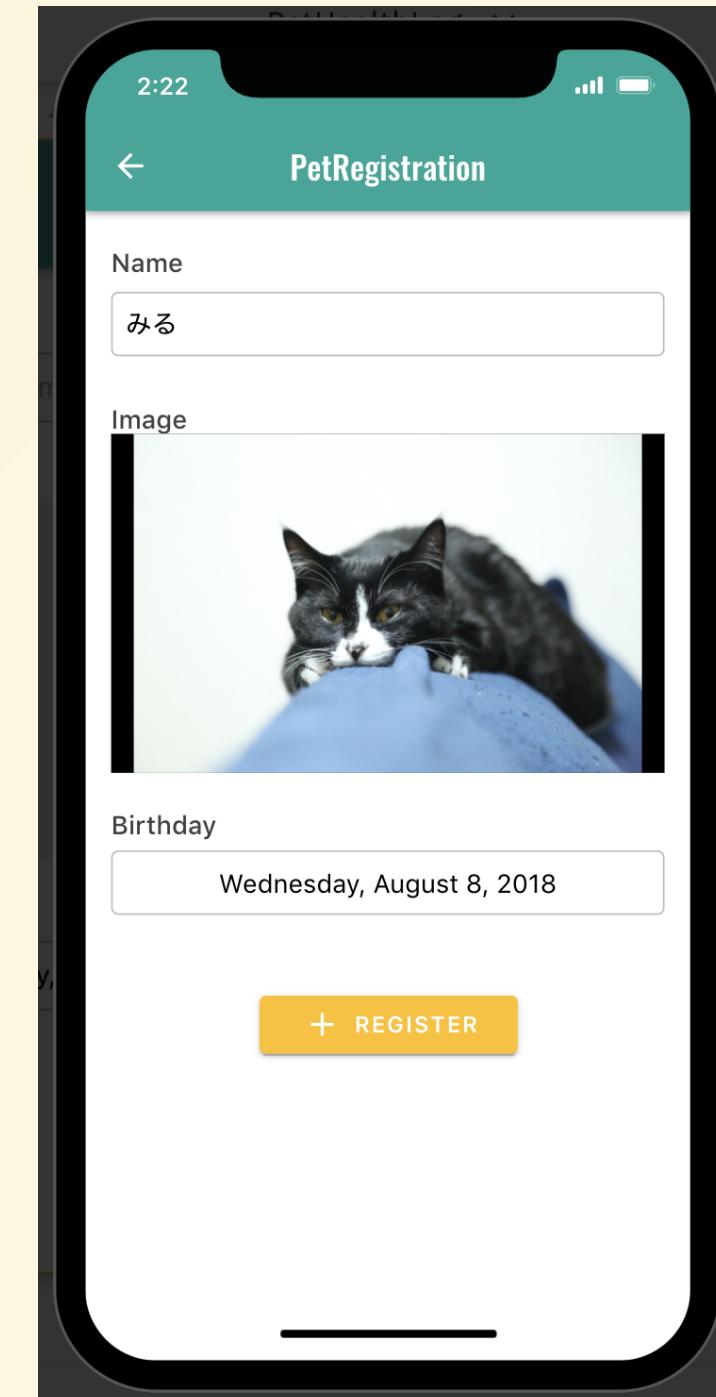
It is recommended that you set up Adalo's database in the same state as the document to avoid confusion in the later work.

# Data manipulation

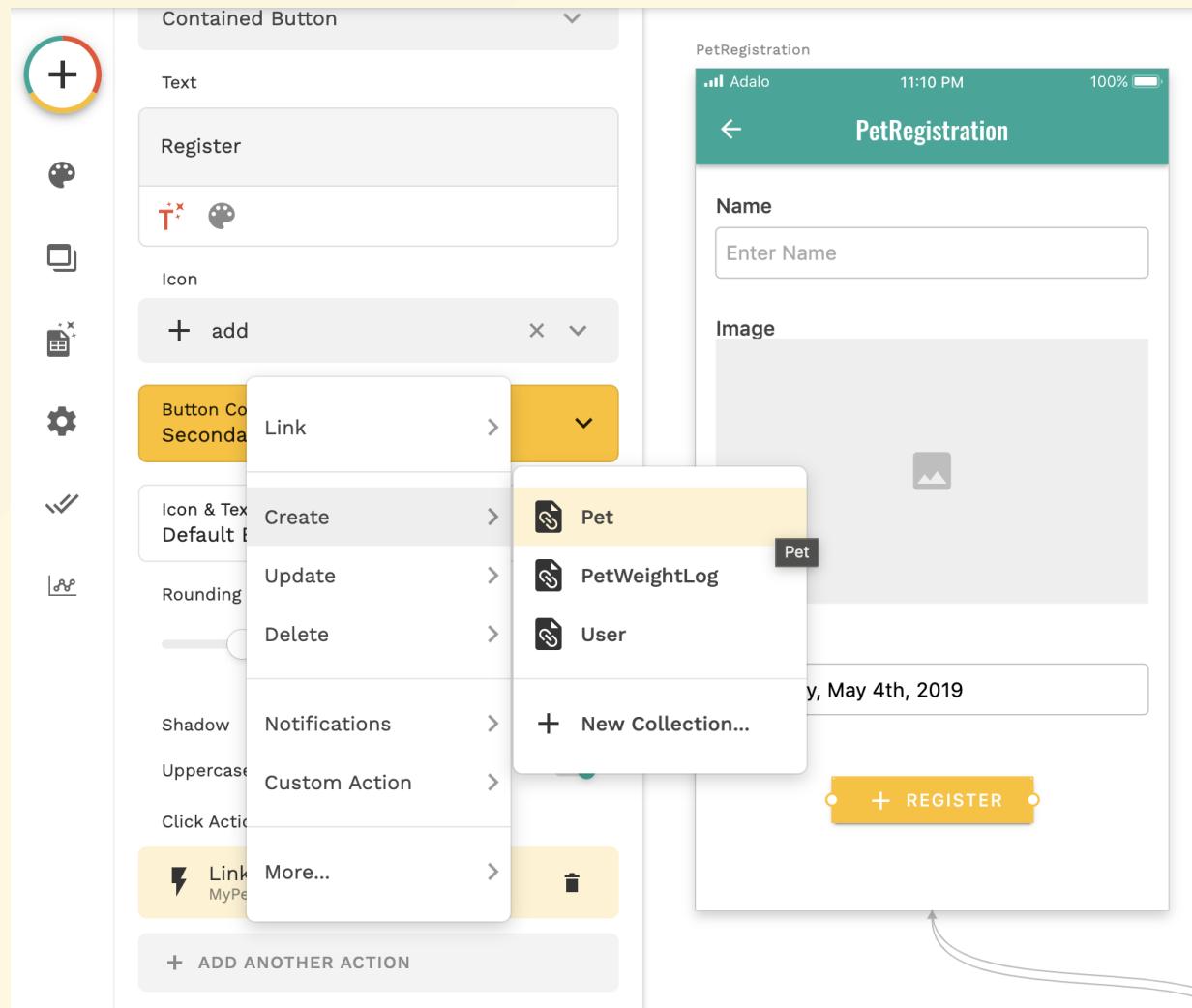
Let's use the database we designed so that we can perform CRUD(CREATE, READ, UPDATE, DELETE) operations in the sample application.

# Creating Data

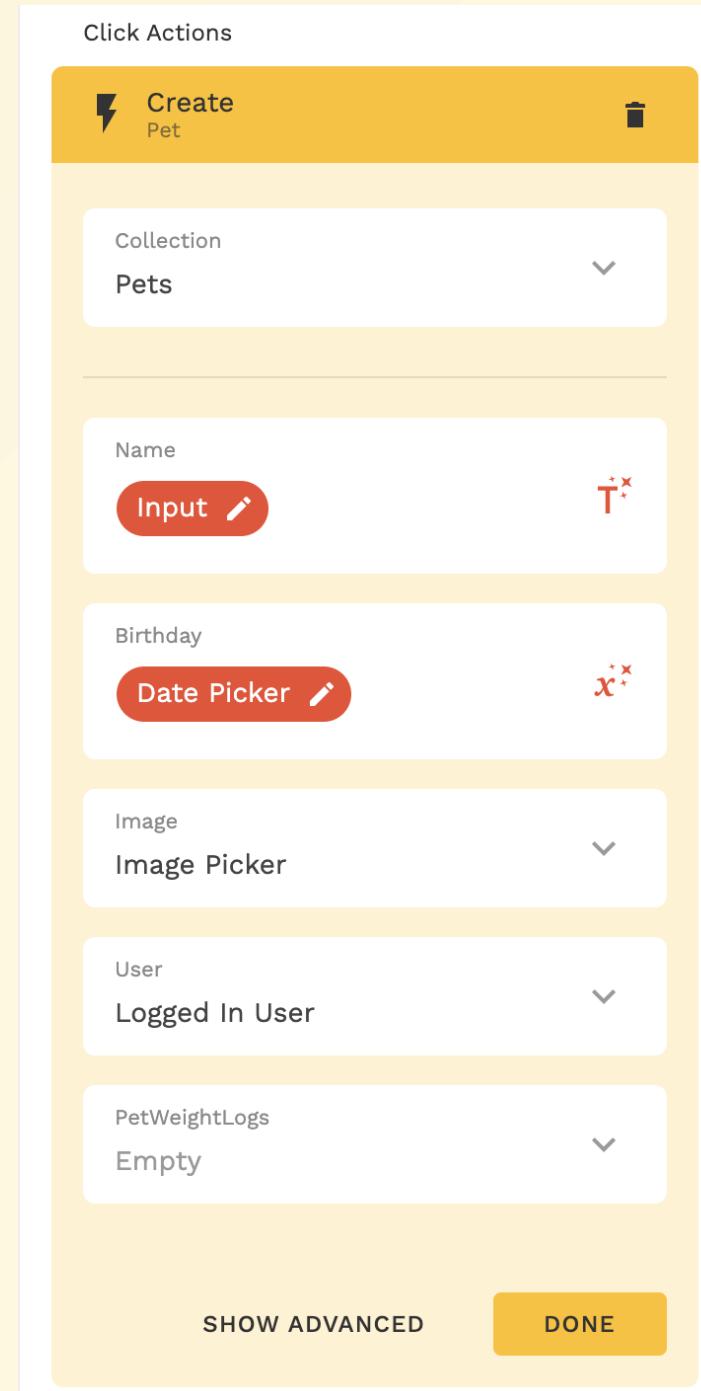
First, let's make it possible to actually register a record of a pet in the pet registration screen.



- Select the "REGISTER" button on the pet registration screen and click "ADD ANOTHER ACTION"
- Select "Create" > "Pet"



- For Name, select "Input" from Other Components.
- For Birthday, select "Date Picker" from Other Components.
- For Image, select "Image Picker" from Other Components.
- For User, select "Logged In User".
- Leave PetWeightLogs as Empty(not required here).



Let's try to register a pet with the Preview function.  
It is OK if you can register record in the Pets Collection

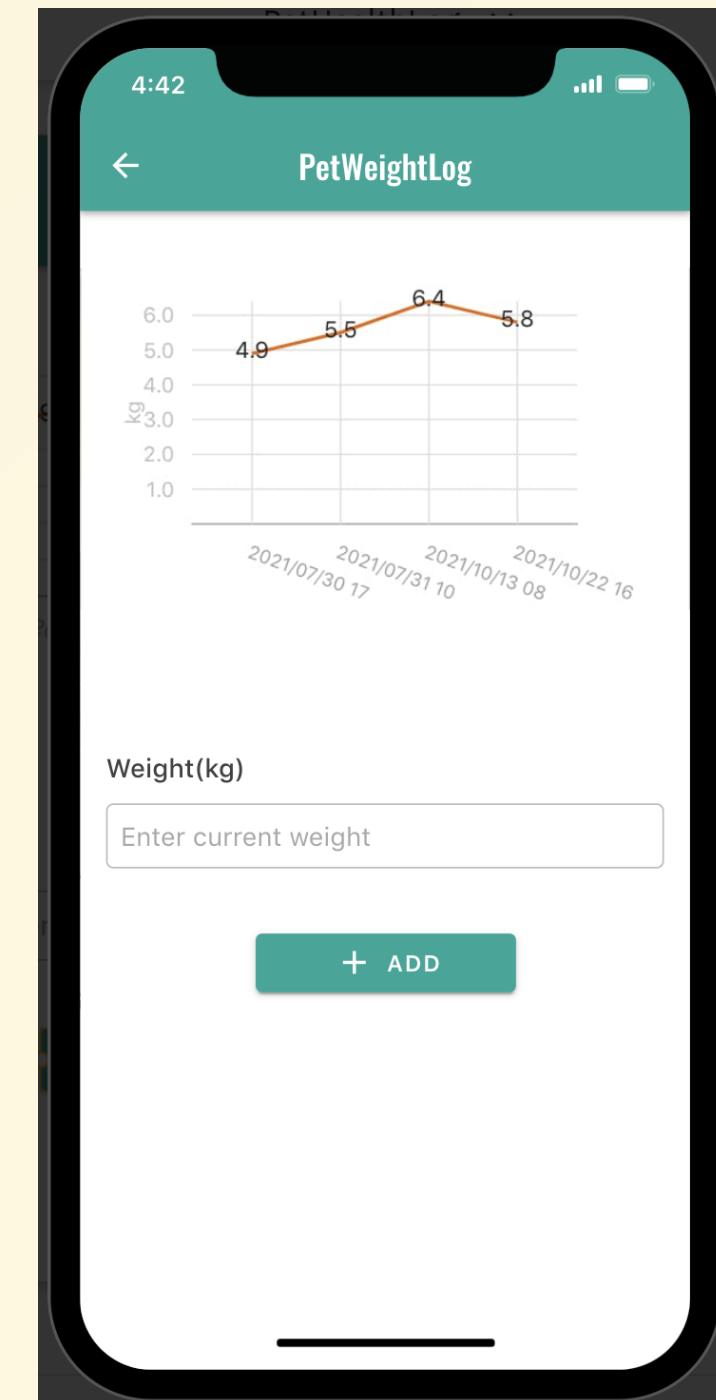
Pets

[+ ADD PET](#)

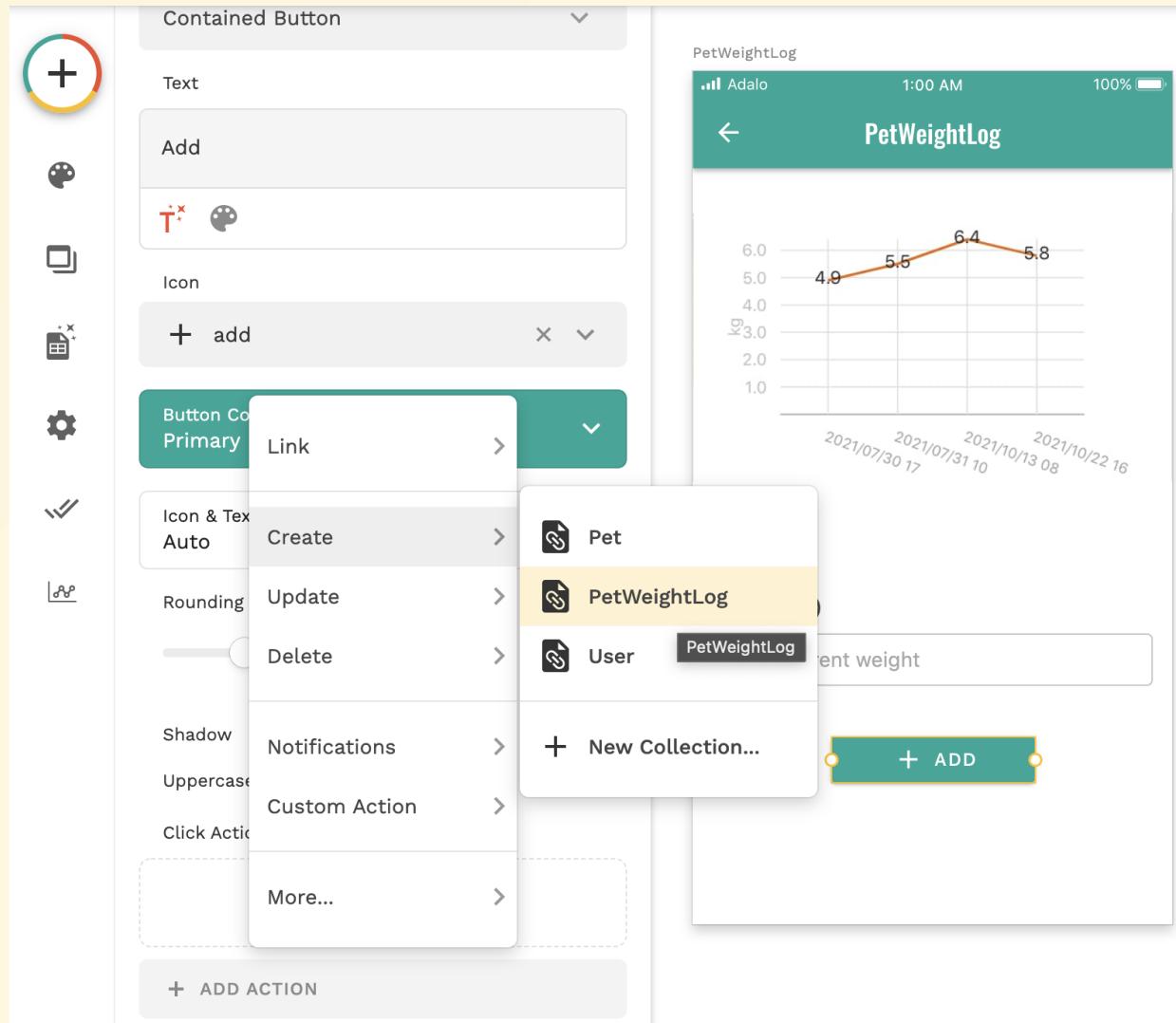
checkbox	Name	Birthday	Image	User	PetWeightLogs	Created
<input type="checkbox"/>	みる(Database Record)	8/8/2018		imahashi@guildworks.jp		43 minutes ago

[DONE](#)

Next, let's make it possible to register the current weight in the pet weight management screen.

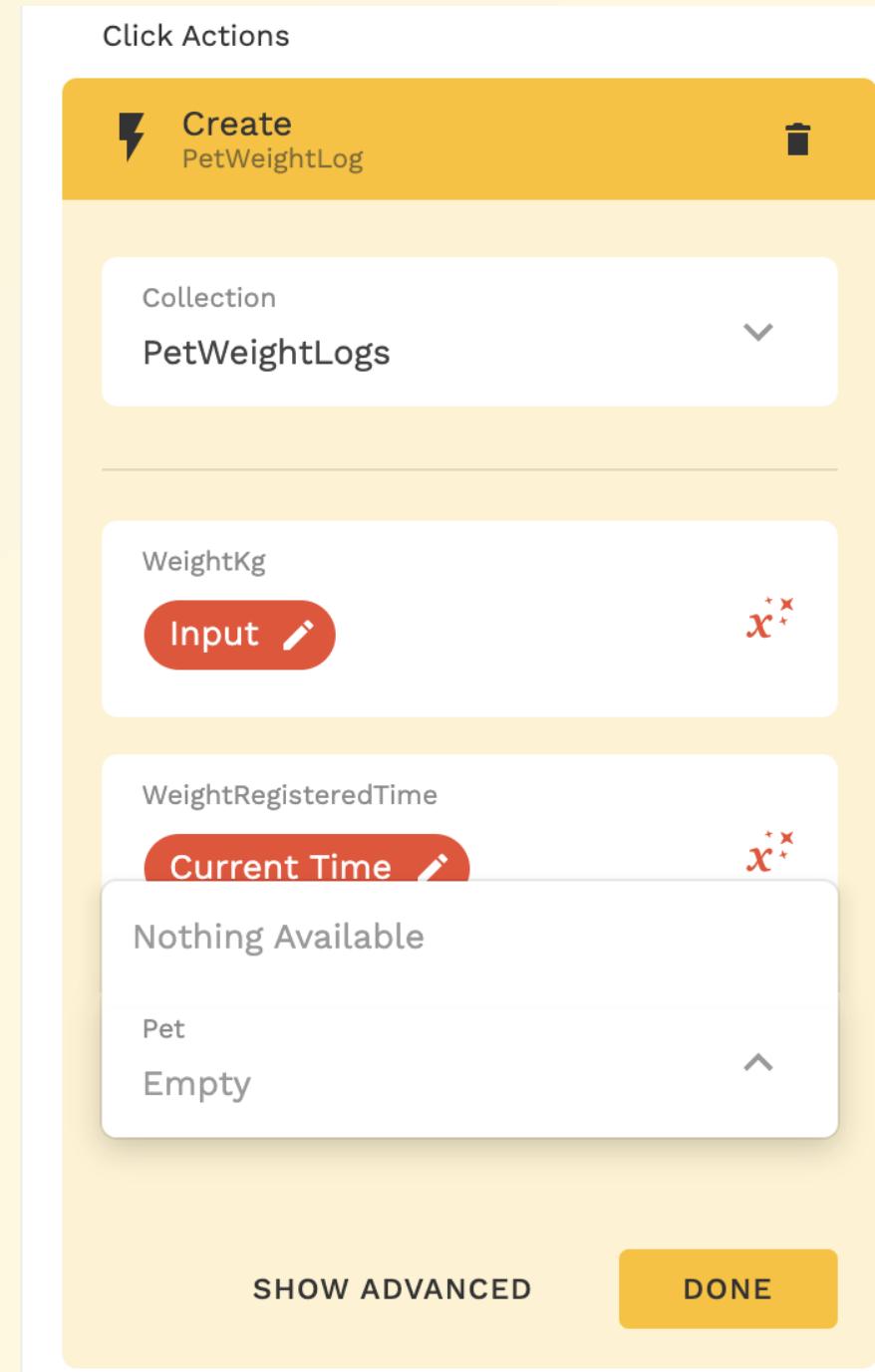


- Select the "ADD" button on the pet weight management screen and click "ADD ACTION".
- Select "Create" > "PetWeightLog"



Enter the following.

- For WeightKg, select "Input" from Other Components.
- For WeightRegisteredTime, select "Date&Time" > "Current Time".
- For Pet, Nothing is Available. Leave it as Empty for now.(We will configure this later)

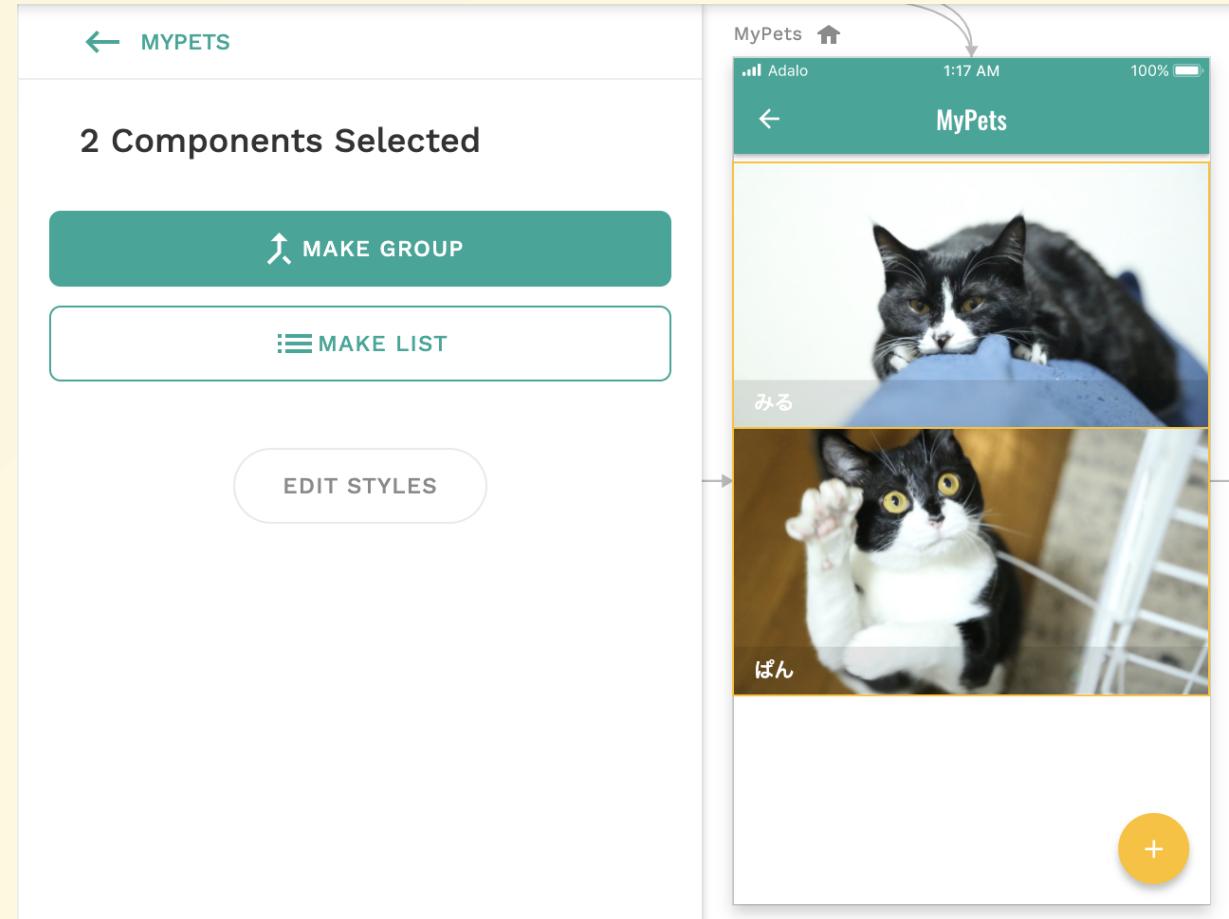


# Displaying Data (READ)

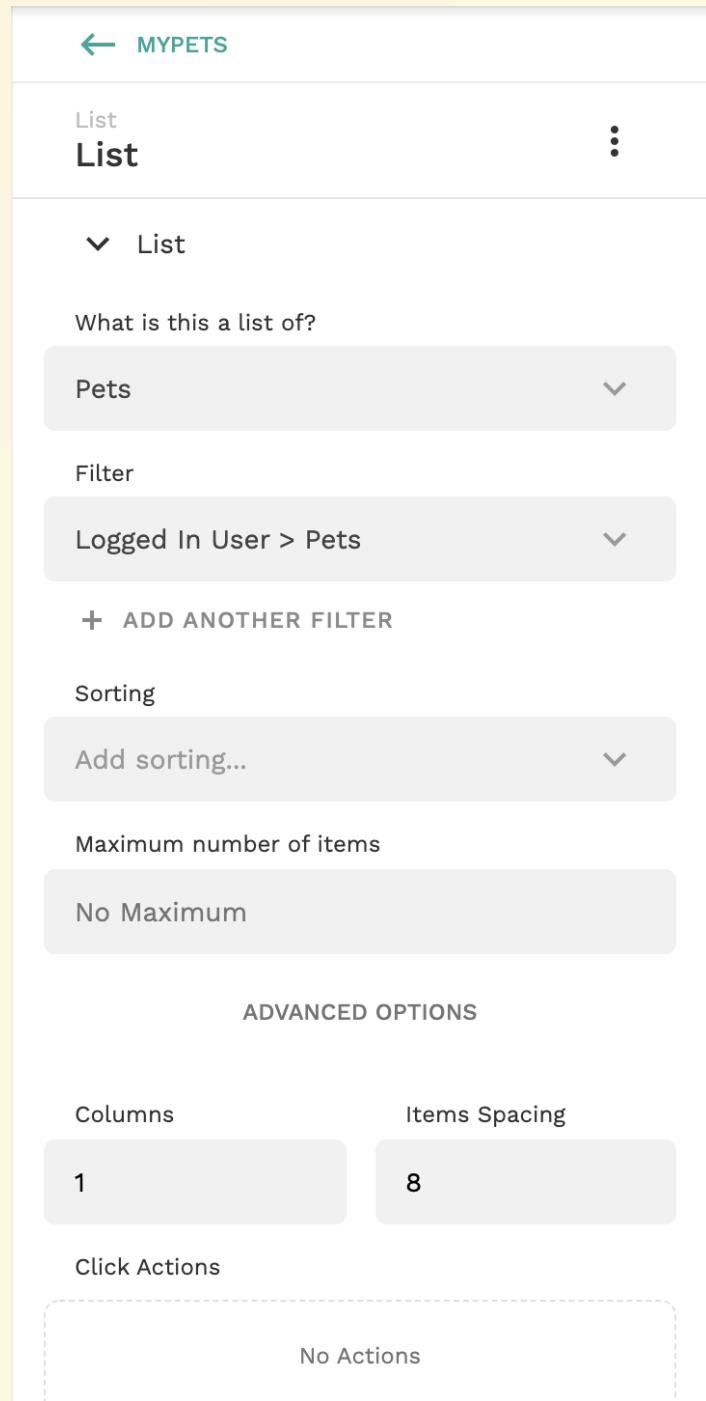
First, let's make registered pets displayed in pets list screen.



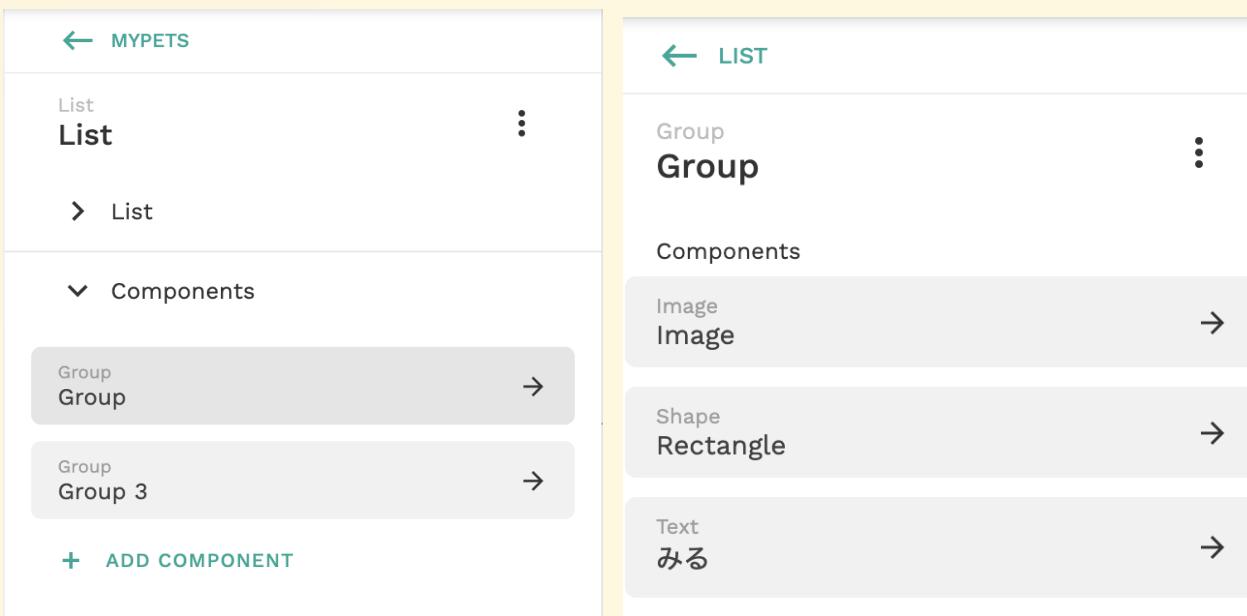
- Select the two components that display the pet's image and name, and click "MAKE LIST".



- Select "Pets" in What is this a list of?
- Select "Logged In User" > "Pets" in Filter

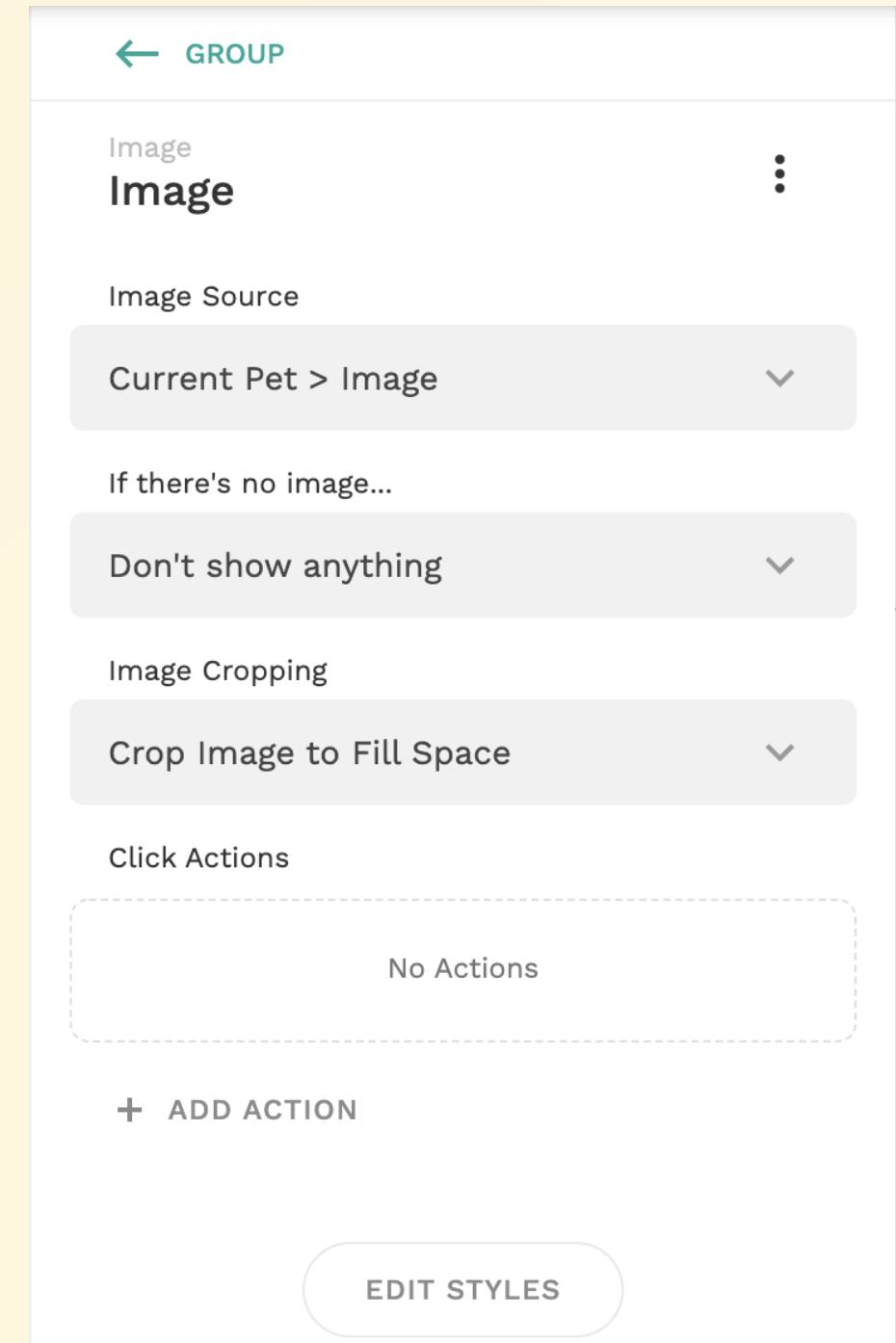


- Click the first Group which corresponds to first pet you manually created.
- Let's edit the Image component and the Pet Name component in this Group.



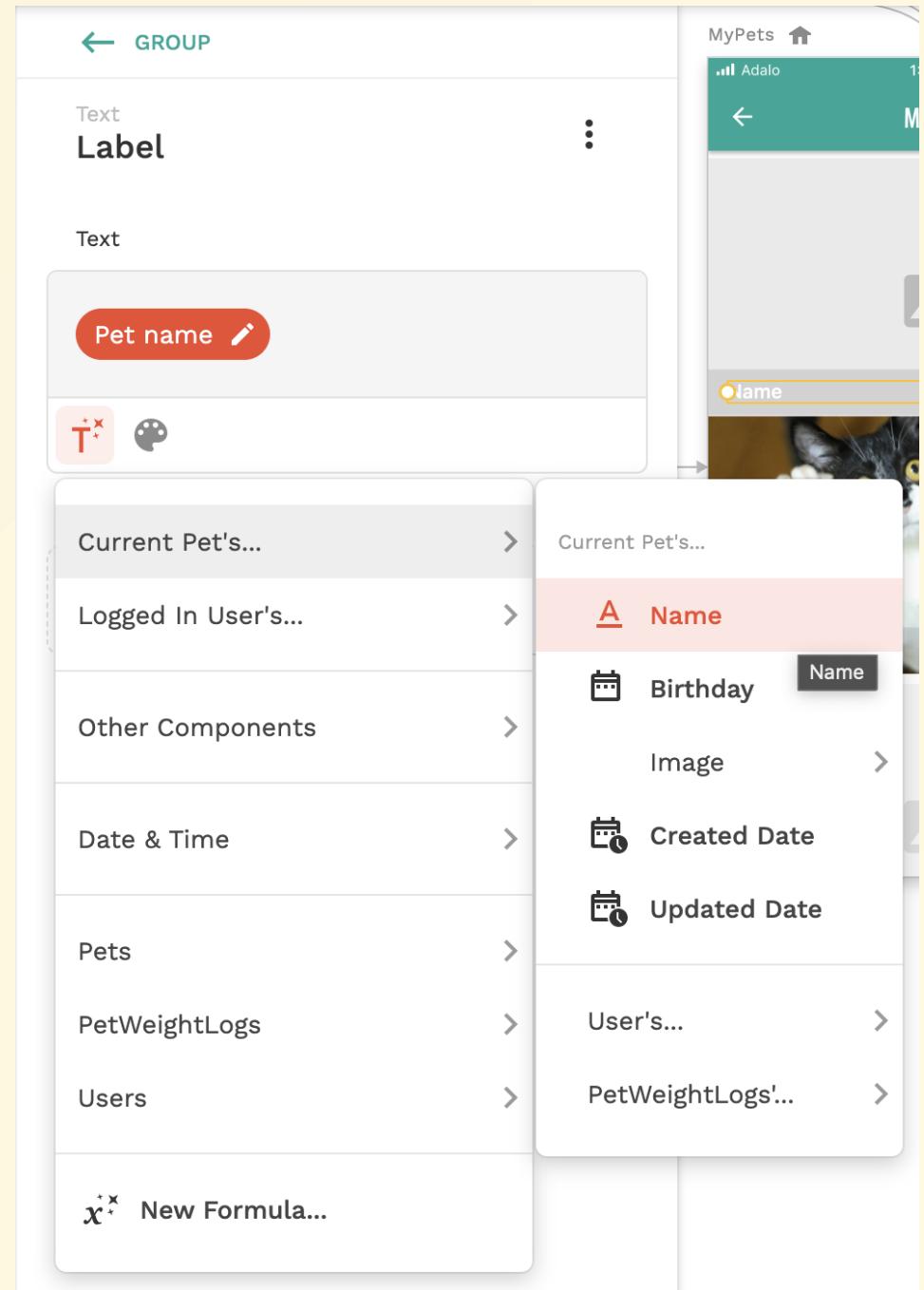
## Edit the Image component

- In Image Source, select "Database" > "Current Pet's" > "Image"
- In "If there's no image...", select "Don't show anything"
  - Alternatively, you can select "Show a place holder image" and set your [pet's silhouette image](#)

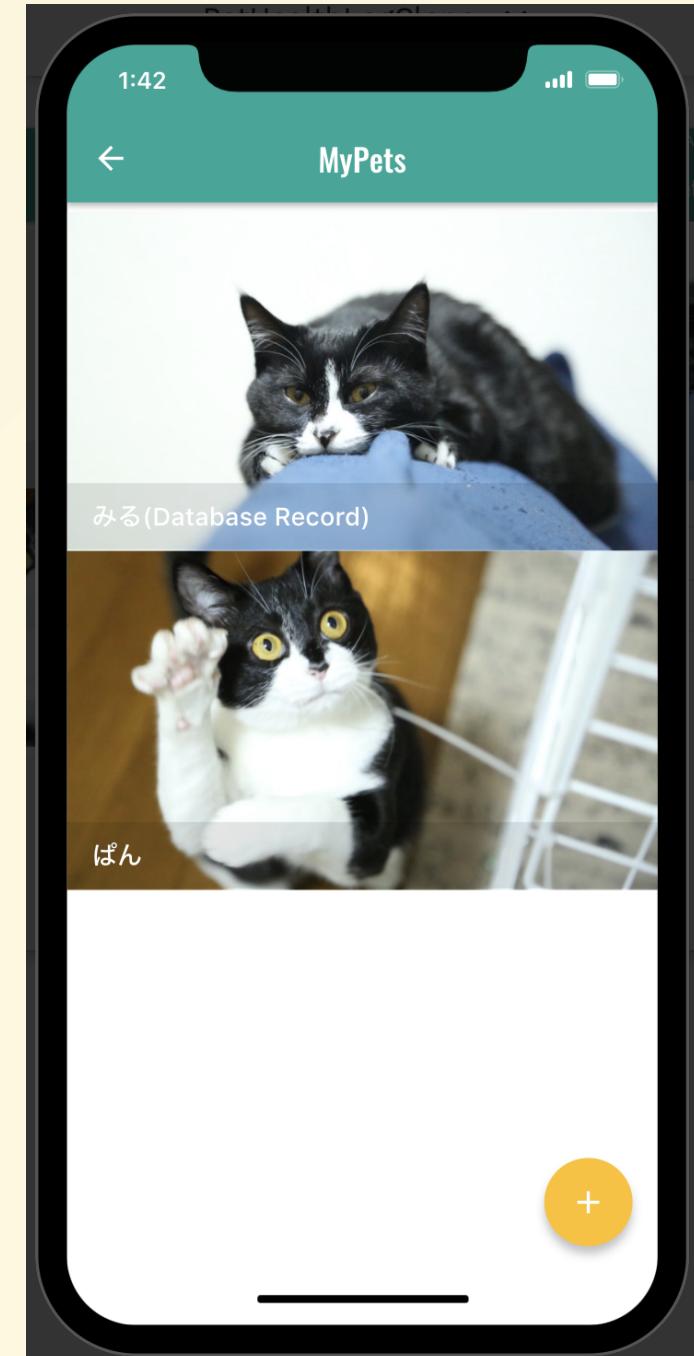


# Edit the pet name component

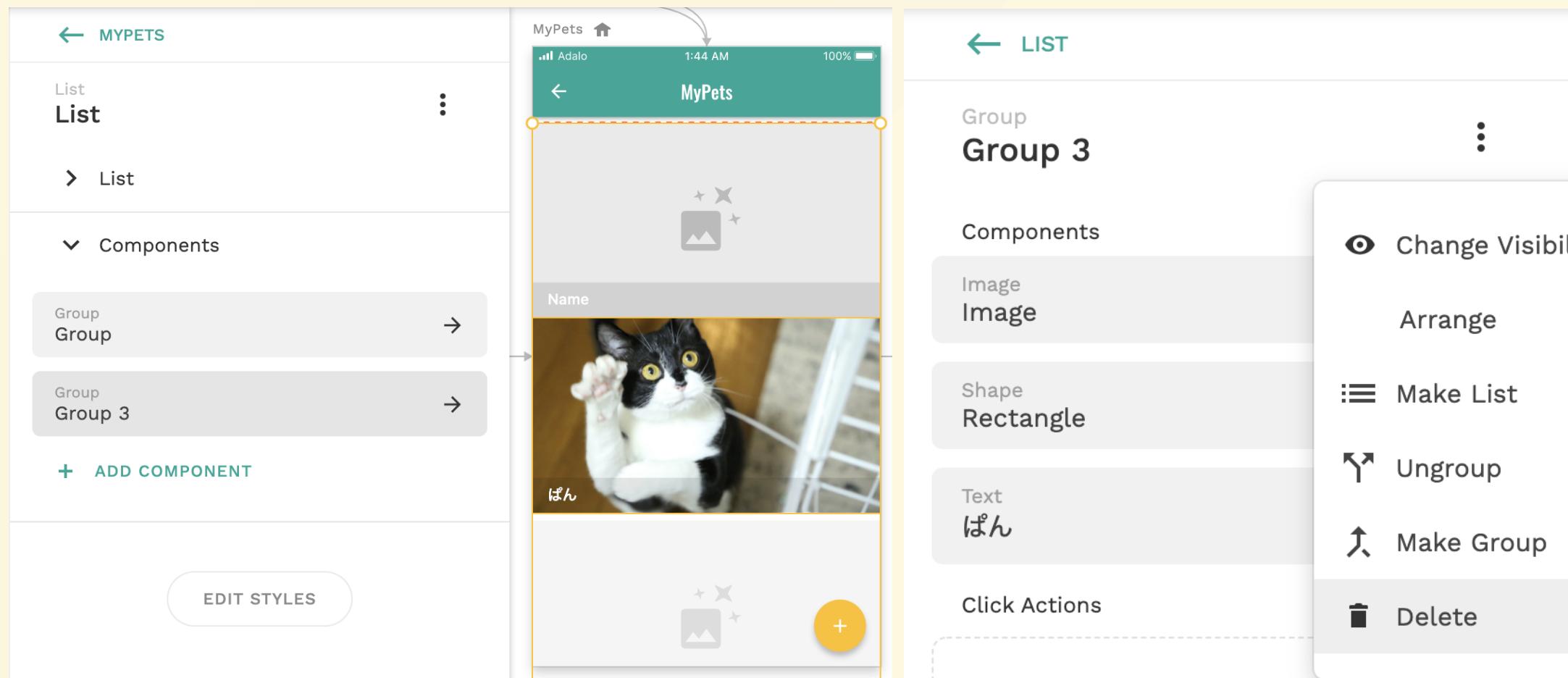
- Under Add Magic Text,  
select "Current Pet's" >  
"Name"



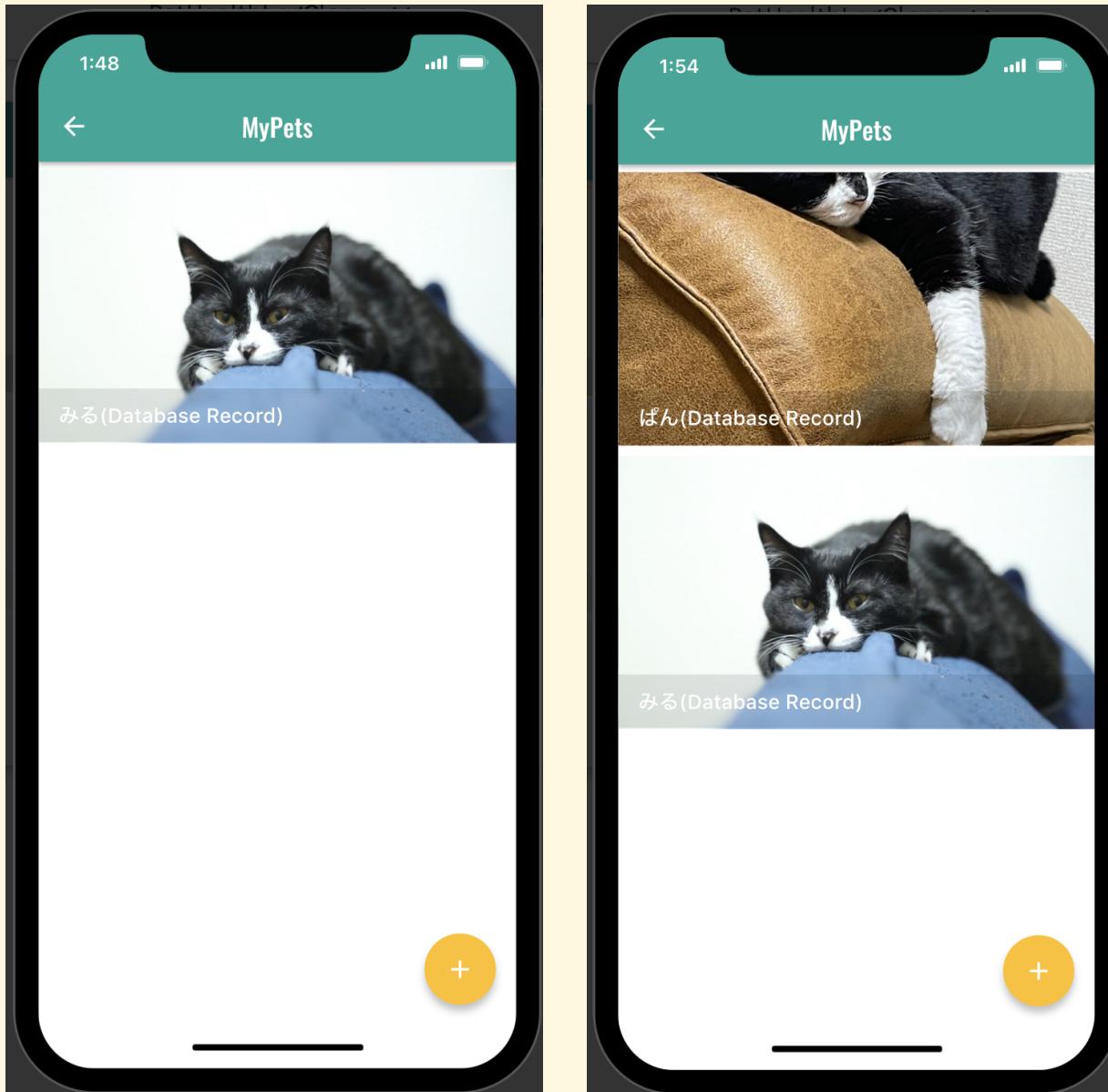
- When you check it by preview function, you will see the Record registered in the database as the first pet.



- Delete the second Group (which displays static second pet) in the component that makes up the List of pets, since it is not needed anymore.

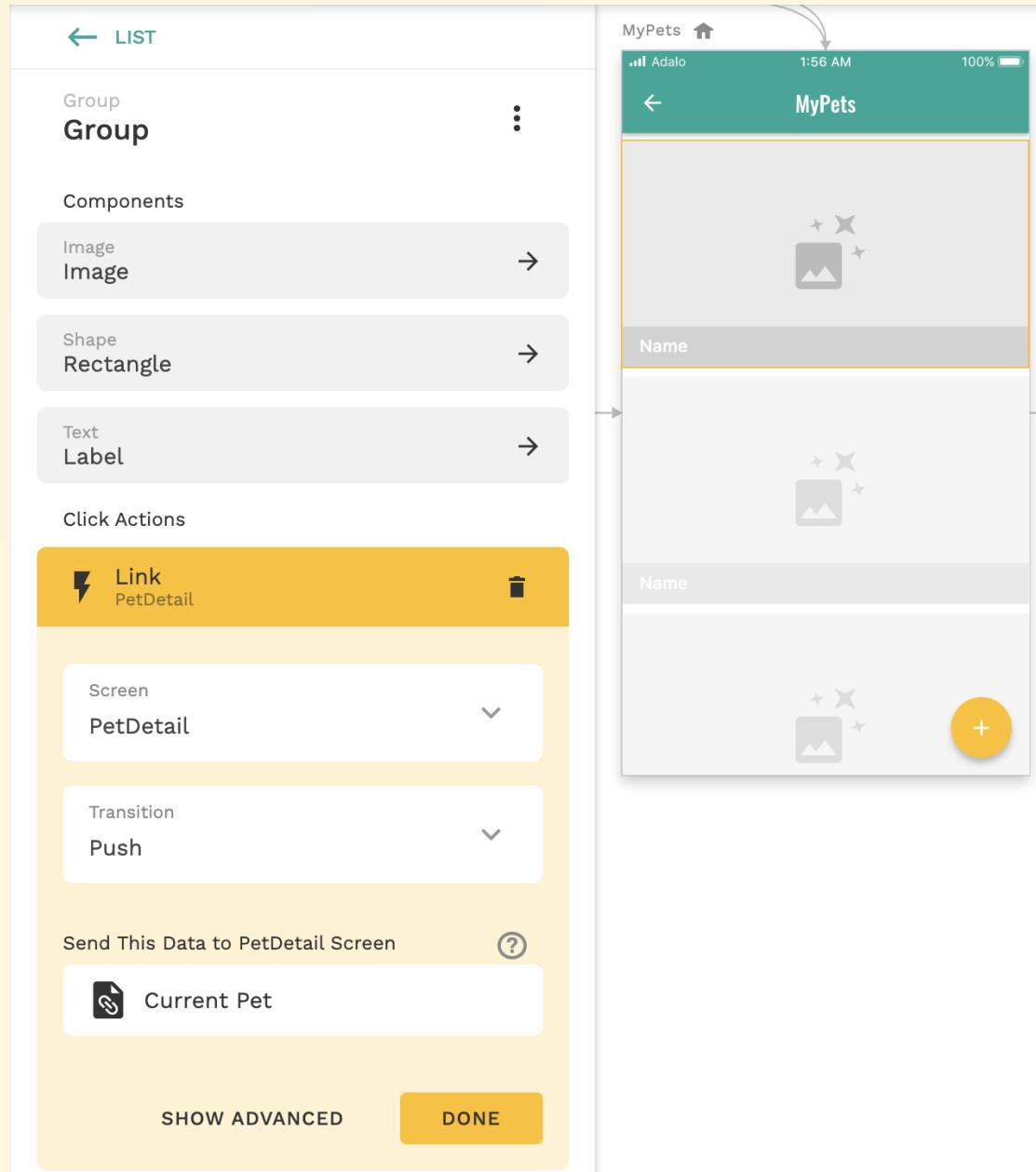


- When you check it by preview function, only the pets you have registered in the database are now displayed.
  - If you register additional pets, multiple pets will be displayed.

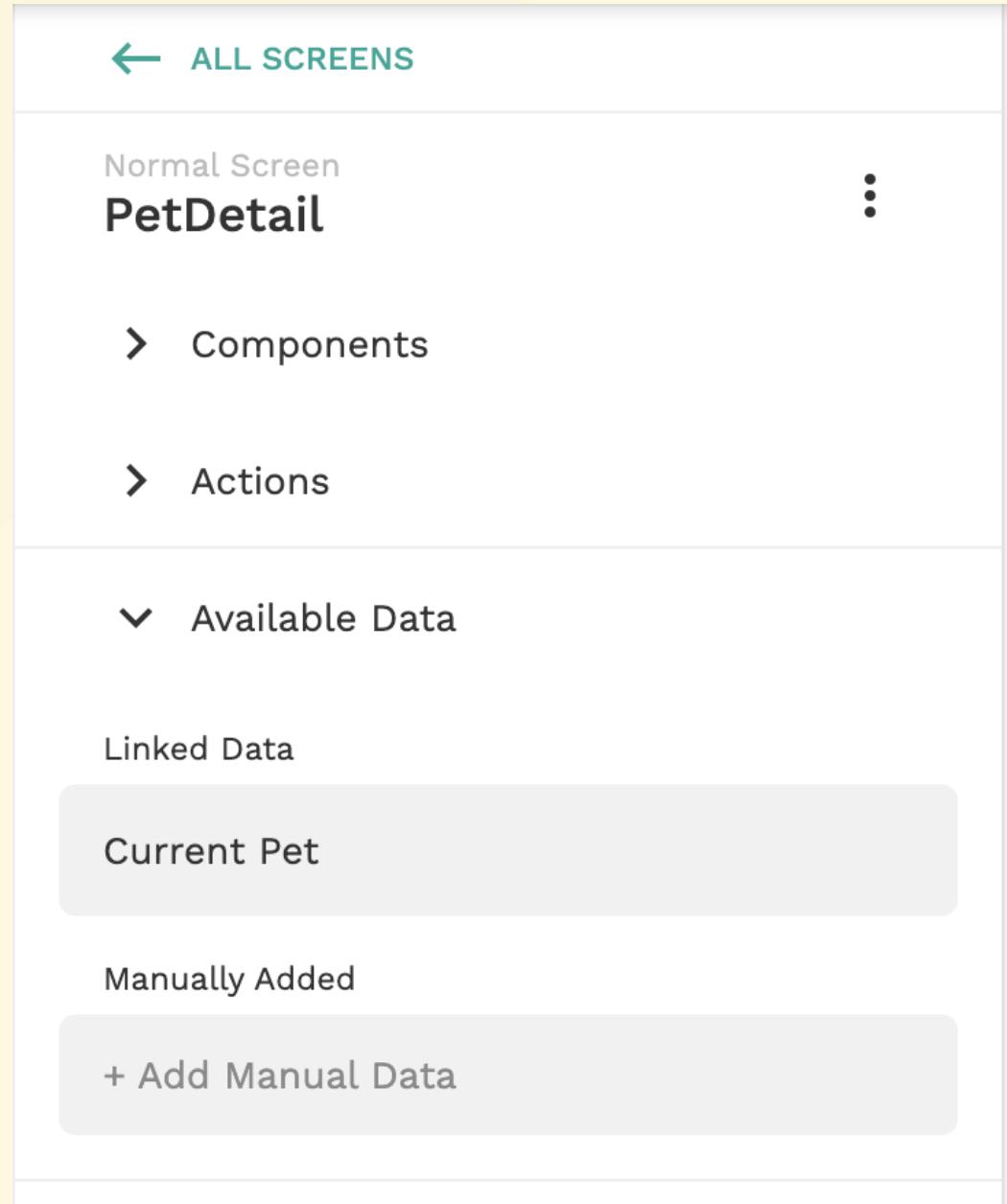


Make sure that when you click on a pet in the pet list, you can go to the detail screen of that pet.

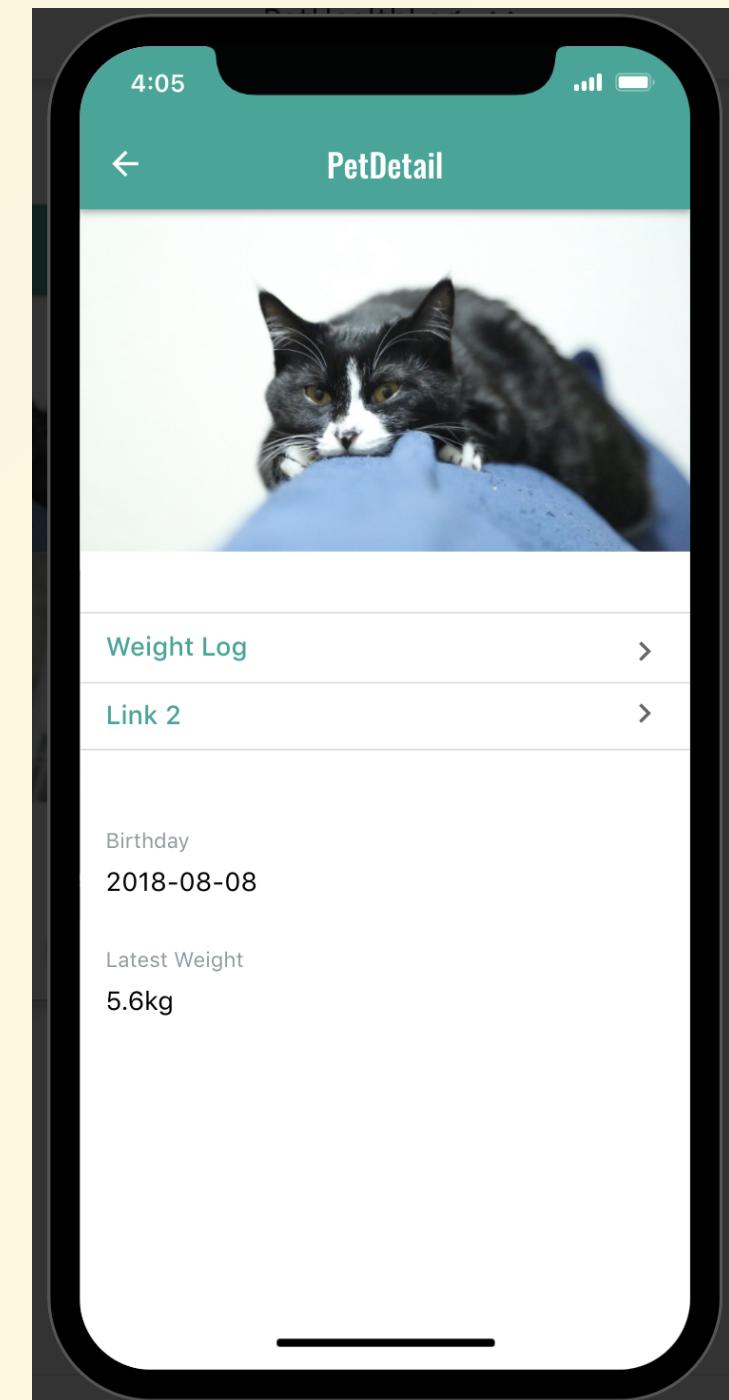
- "Current Pet" is automatically set in the "Send This Data to PetDetail Screen" of the Link set in the Group component of the pet.



- Since "Current Pet" was set in "Send This Data to PetDetail Screen" of Link from Pet List Screen, "Current Pet" is set as "Linked Data" in "Available Data" of Pet Detail Screen.
  - As a result, the pet selected in the Pet List screen (Current Pet) can be handled in the Pet Detail screen.

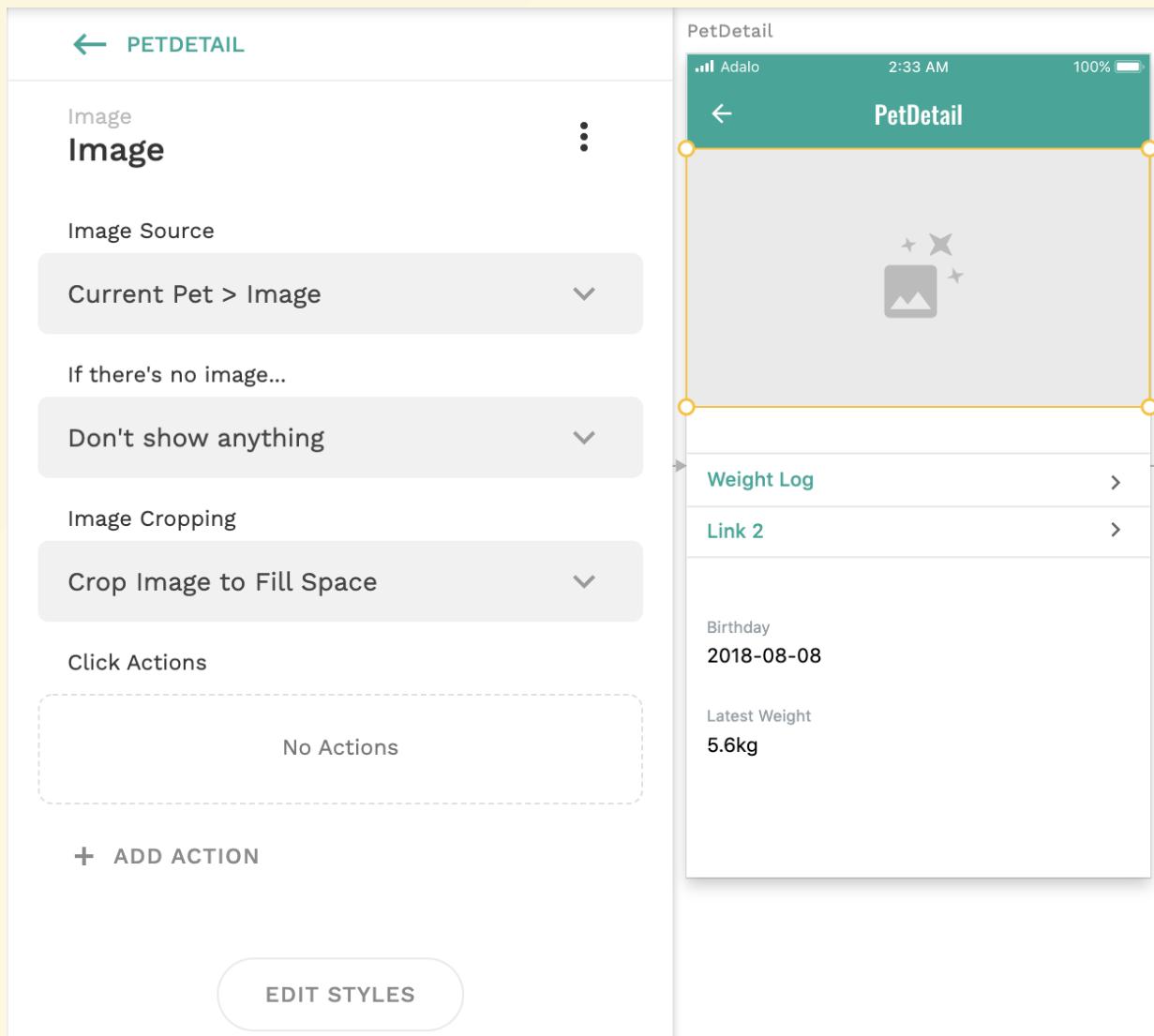


Next, we will make sure that the pet selected in the Pet List screen is displayed in the Pet Detail screen.



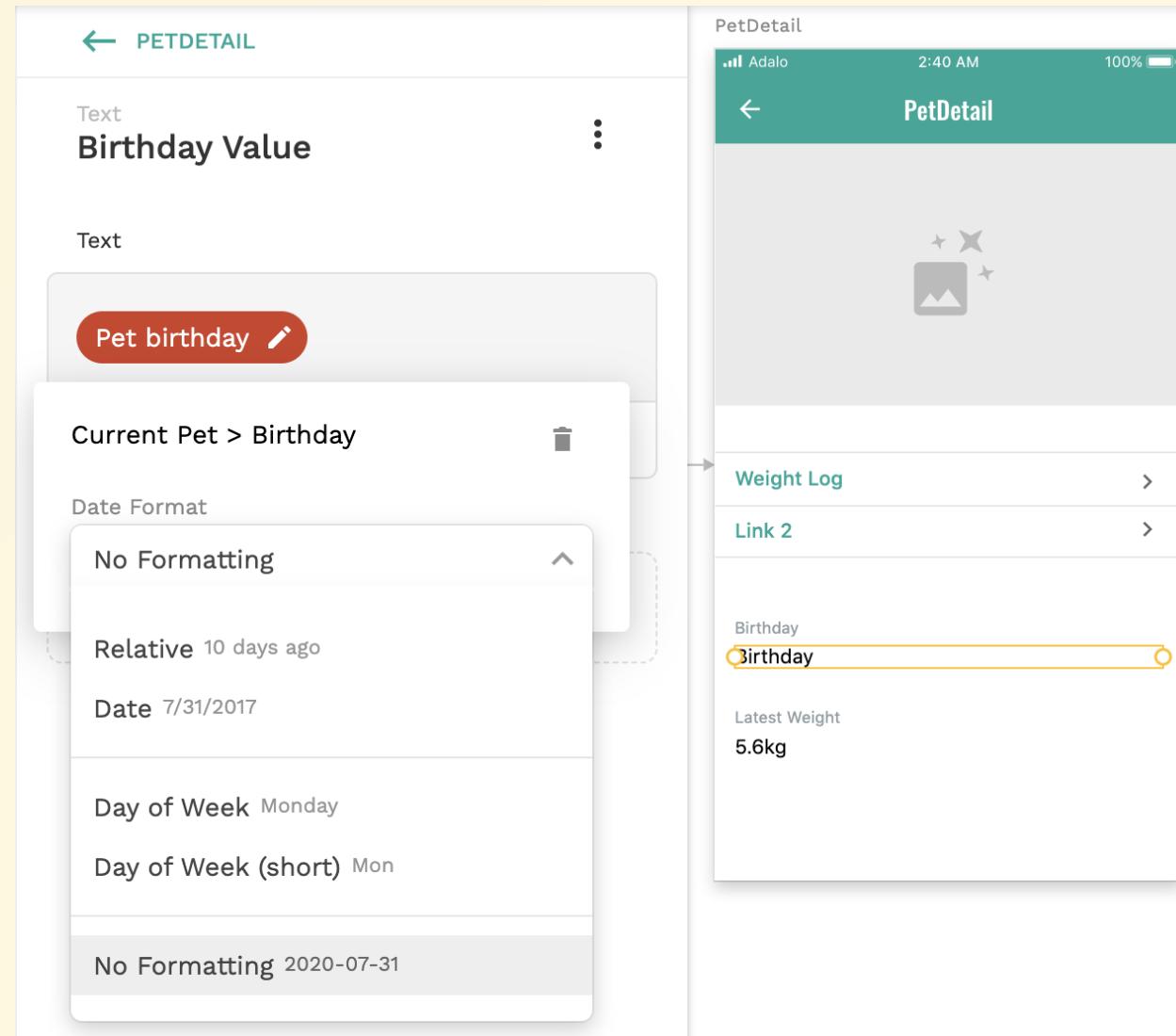
Click on the Image component and configure it as follows.

- In Image Source, select "Database" > "Current Pet's" > "Image".
- In "If there's no image...", select "Don't show anything"
  - Alternatively, you can select "Show a place holder image" and set your [image](#)



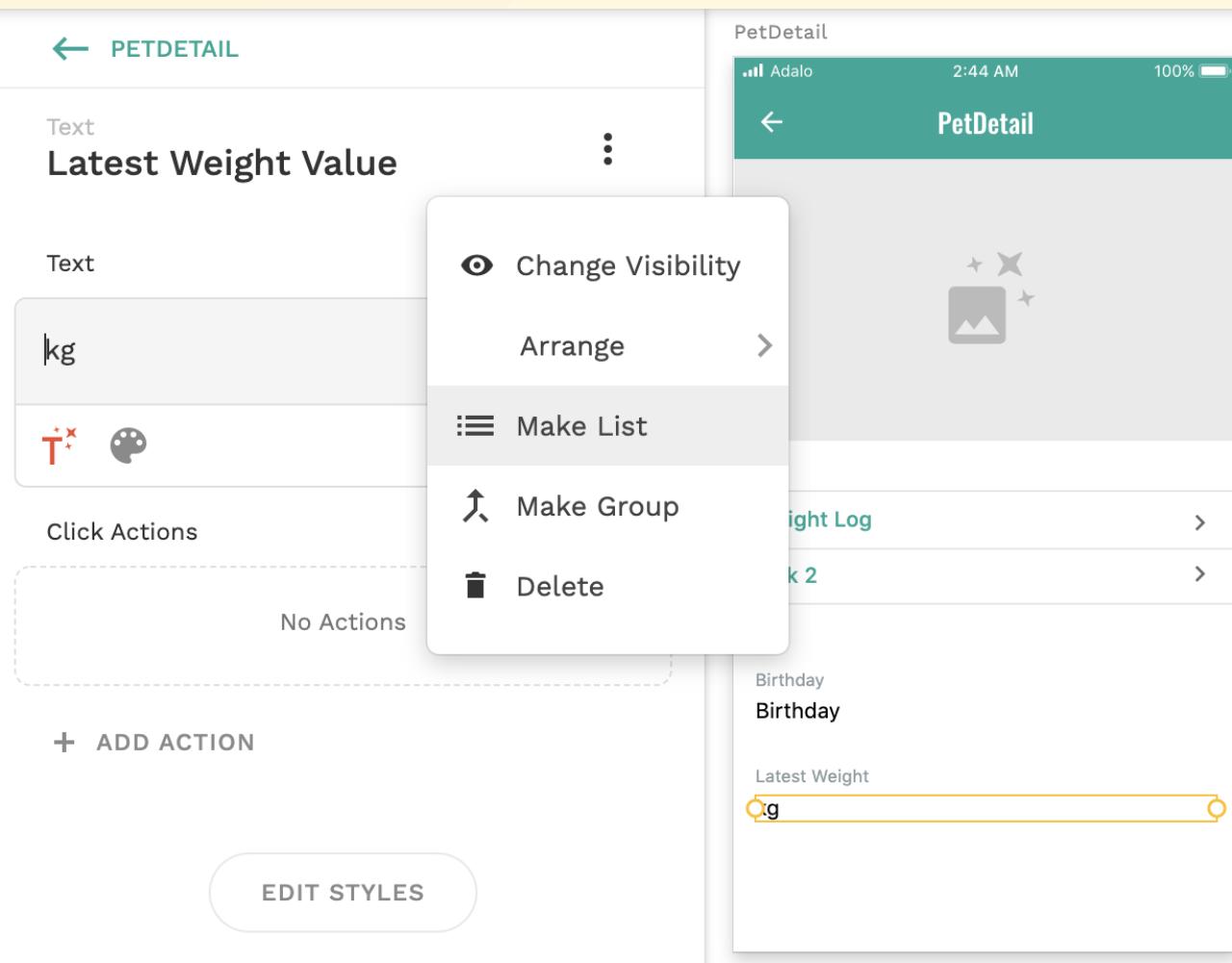
Click the Birthday Value component and configure it as follows.

- Select "Current Pet's" > "Birthday" in Text
- Select "No Formatting" in Date Format



- Click the Latest Weight Value component and "MAKE LIST".

This is required to displaying the latest weight.  
(The settings on the next page will narrow down the list to the most recent one.)

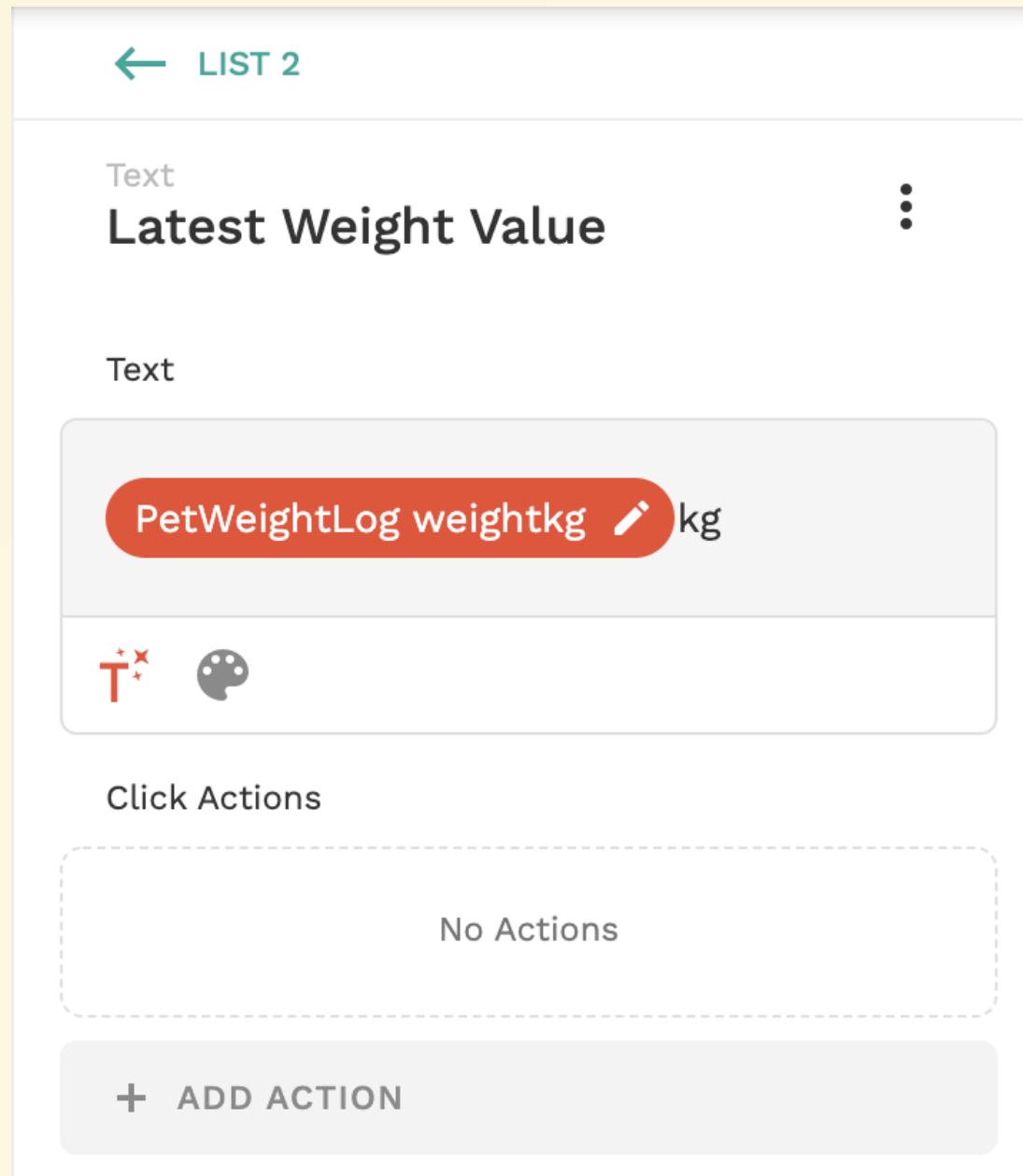


- Select "etWeightLogs" in "What is this a list of?"
- Select "Current Pet" > "PetWeightLogs" in Filter
- In Sorting, select "WeightRegisteredTime - Newest to Oldest"
- Set "Maximum number of items" to 1

This will narrow down the list to only the latest one.

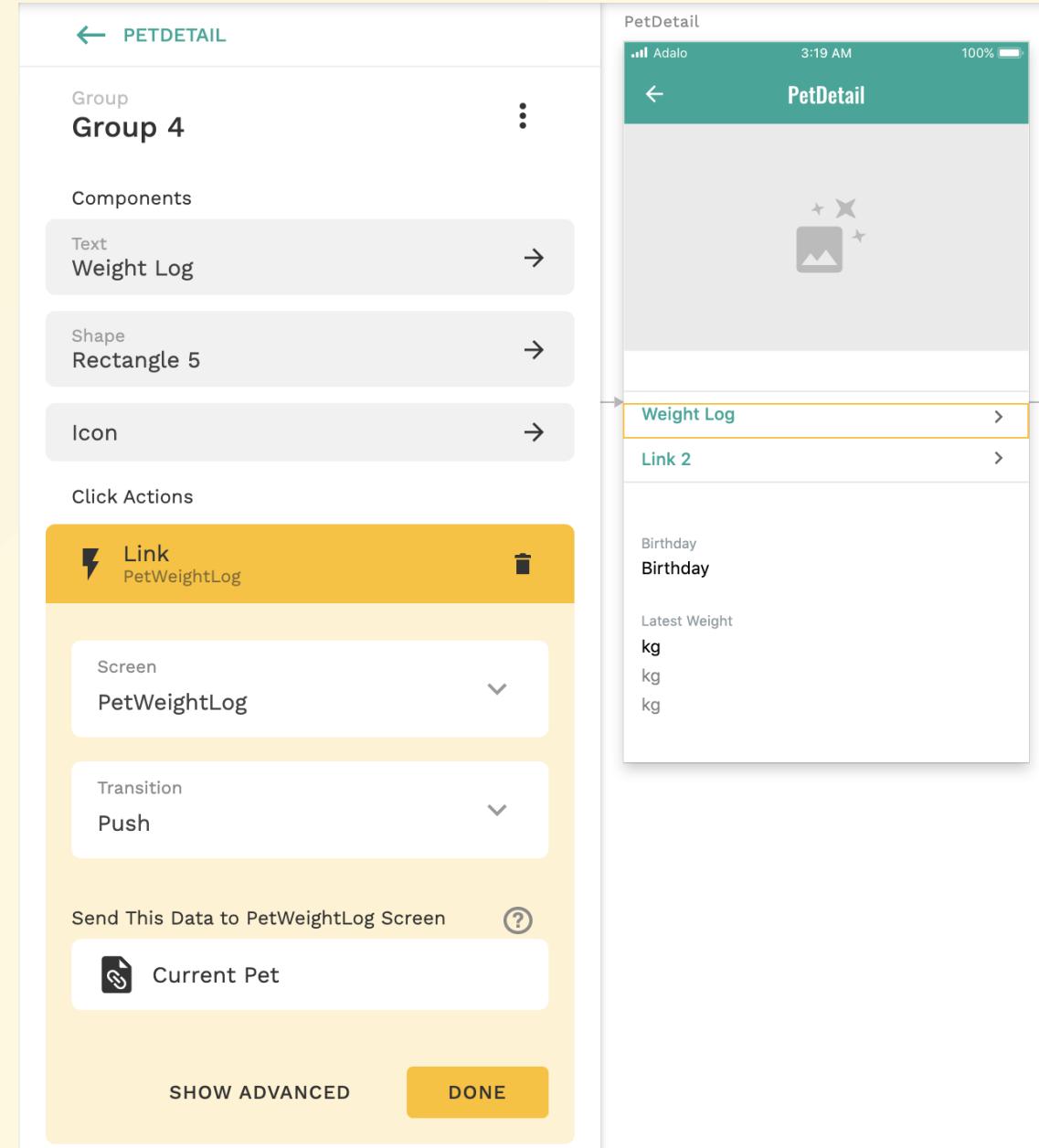
The screenshot displays the Adalo mobile application interface. On the left, the configuration screen for a list named "List 2" is shown. It includes fields for "What is this a list of?", "Filter" (set to "Current Pet > PetWeightLogs"), "Sorting" (set to "WeightRegisteredTime - Newest to Oldest"), and "Maximum number of items" (set to 1). Advanced options like "Columns" (1) and "Items Spacing" (8) are also visible. On the right, the resulting "PetDetail" screen is displayed, showing a pet's profile with a "Weight Log" section. The "Weight Log" section shows a single entry: "Latest Weight" with values "kg", "kg", and "kg" in a dashed red box, indicating it is the latest log.

- Click the Latest Weight Value component, add select "Current PetWeightLog's" > "WeightKg" to the Text, then add "kg" to the end.



Make sure that the selected pets can be passed on when moving from the Pet Detail screen to the Pet Weight Management screen.

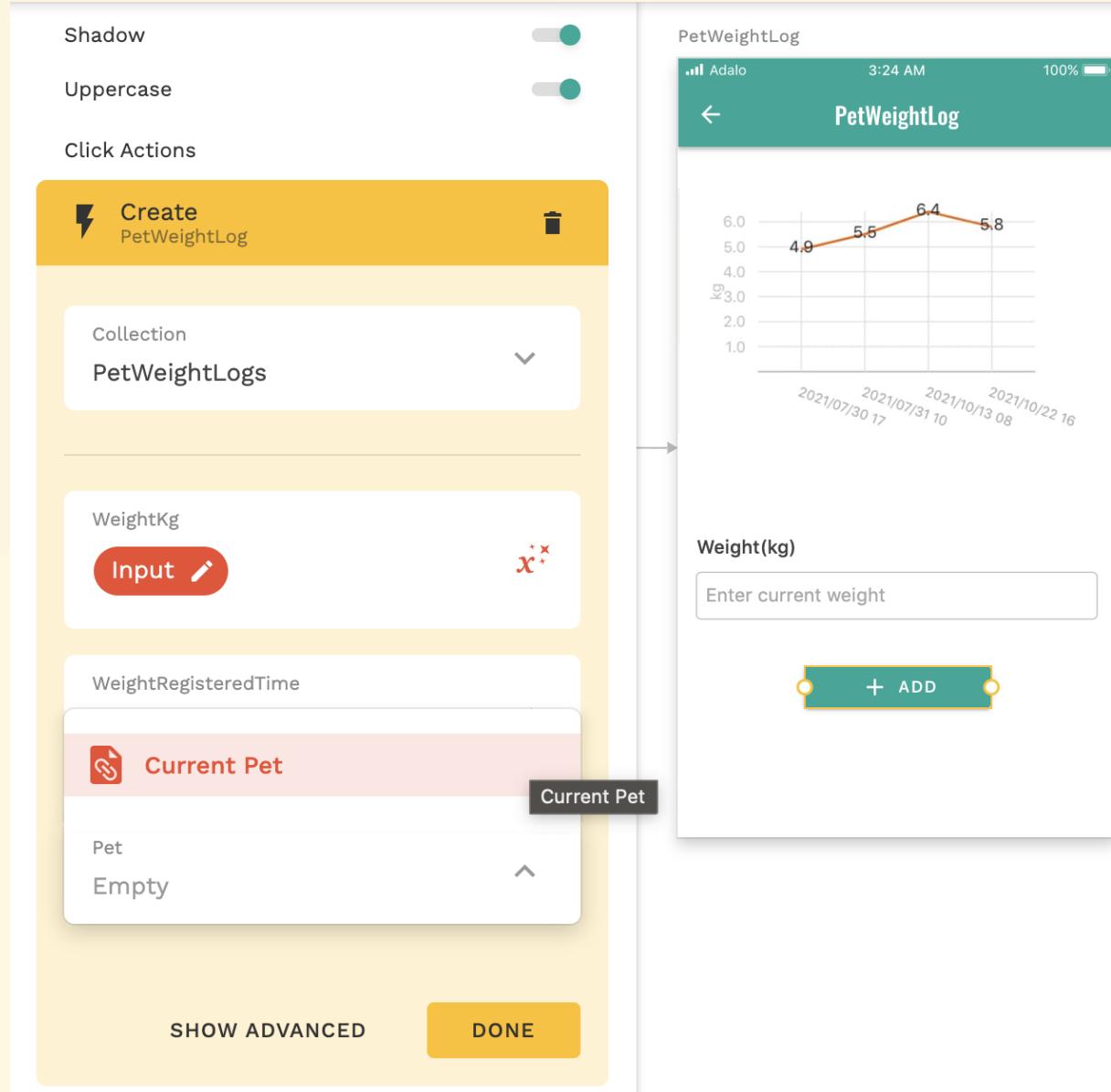
- Select the component of the link "Weight Log", and confirm that "Current Pet" is automatically set in "Send This Data to PetDetail Screen".



(The remaining work of creating data)

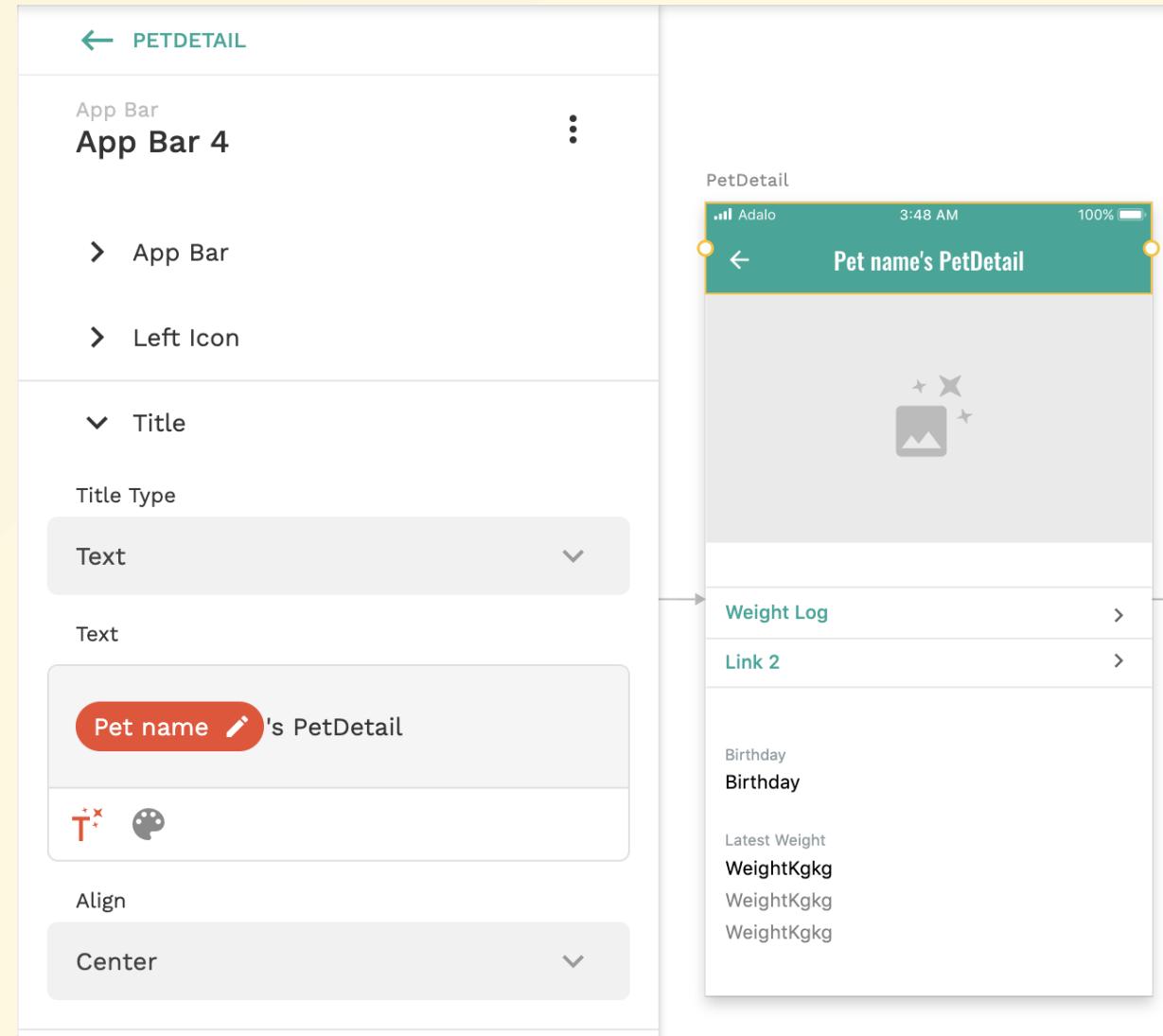
Allow added weight in the Pet Weight Management screen to be registered in the database.

- Select the "ADD" button, and set "Current Pet" to the "Pet" (that was once left as Empty).



Display the name of the selected pet in the header of the Pet Details screen and the Pet Weight Management screen. On each screen,

- Select the App Bar component at the top of the screen and add "Current Pet's" > "Name" to the Title Text.
- Add the string "'s PetDetail" after it.

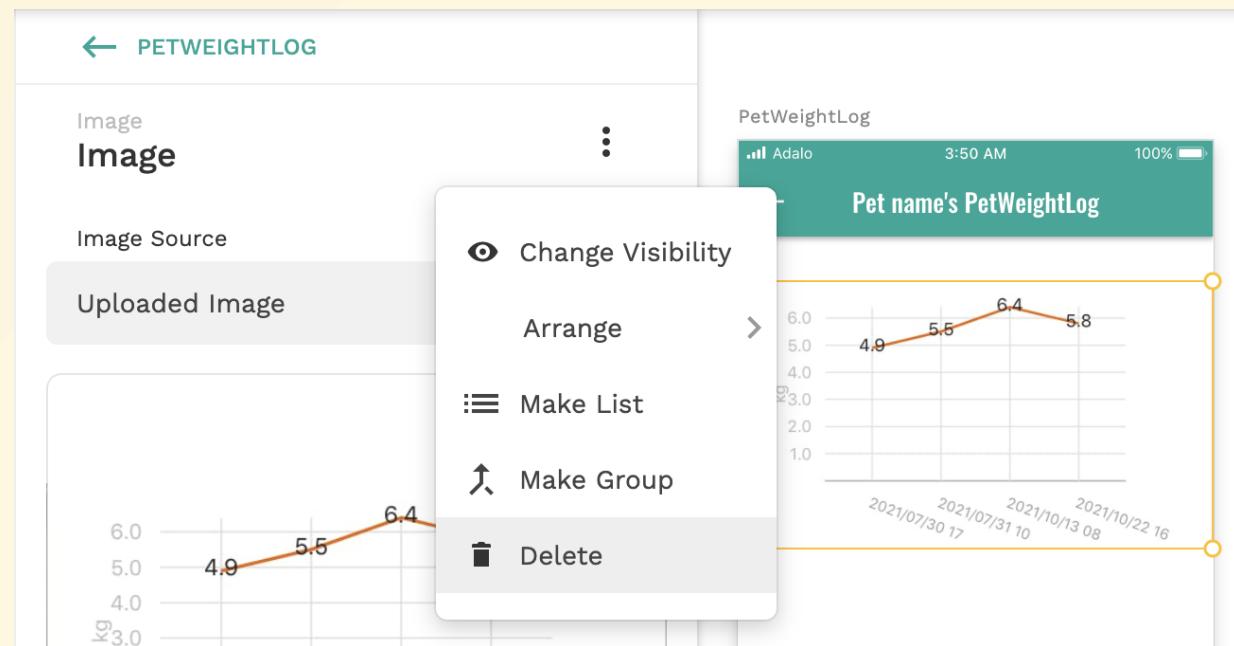


In the Preview function, check the following.

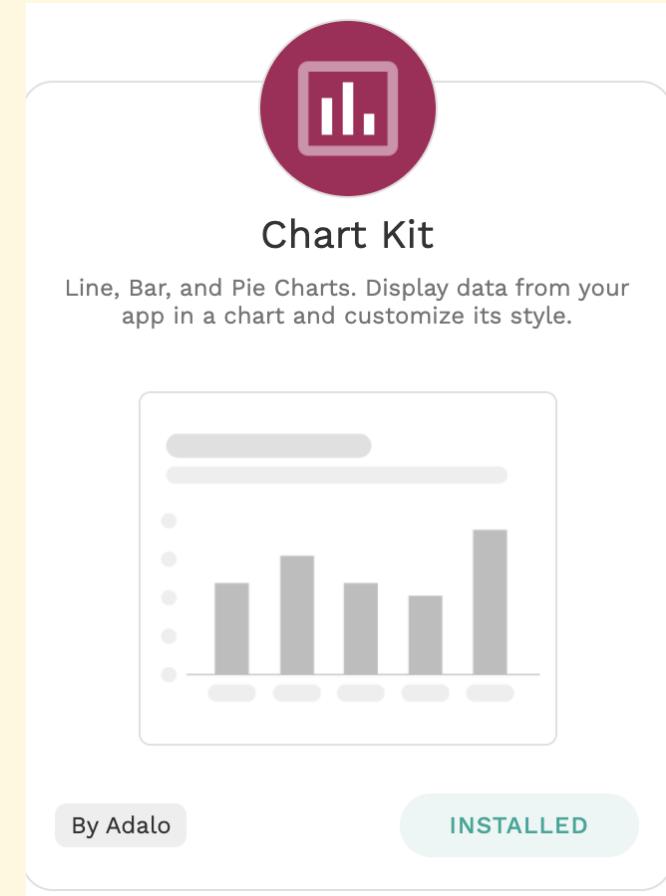
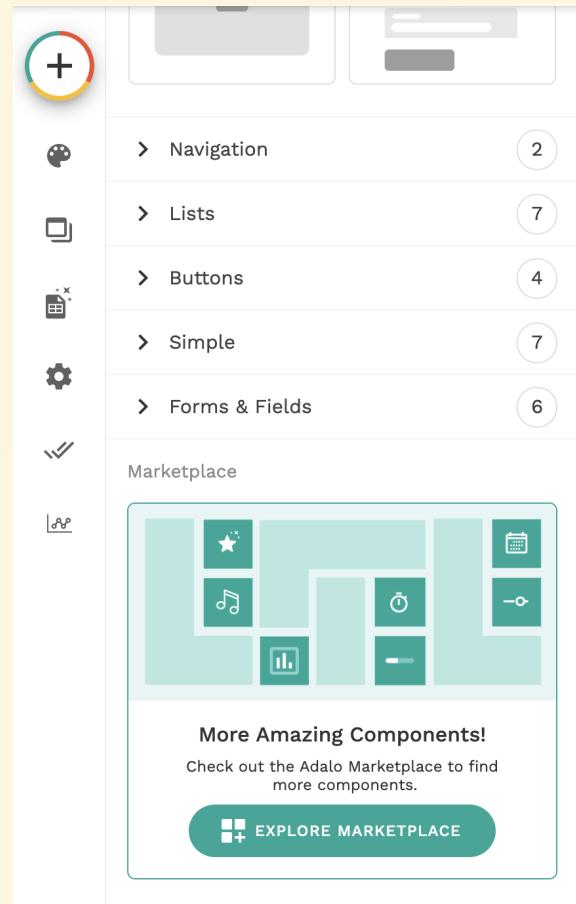
- You can move to the detail screen of the selected Pet from the Pet List screen.
- It is possible to move to the weight management screen of the selected pet from the pet detail screen.
- When the pet weight is registered, the record is registered in the PetWeightLogs Collection of the database.
- The latest weight is displayed on the pet detail screen.

Next, we will display registered pet's weight logs as a graph in the pet weight management screen.

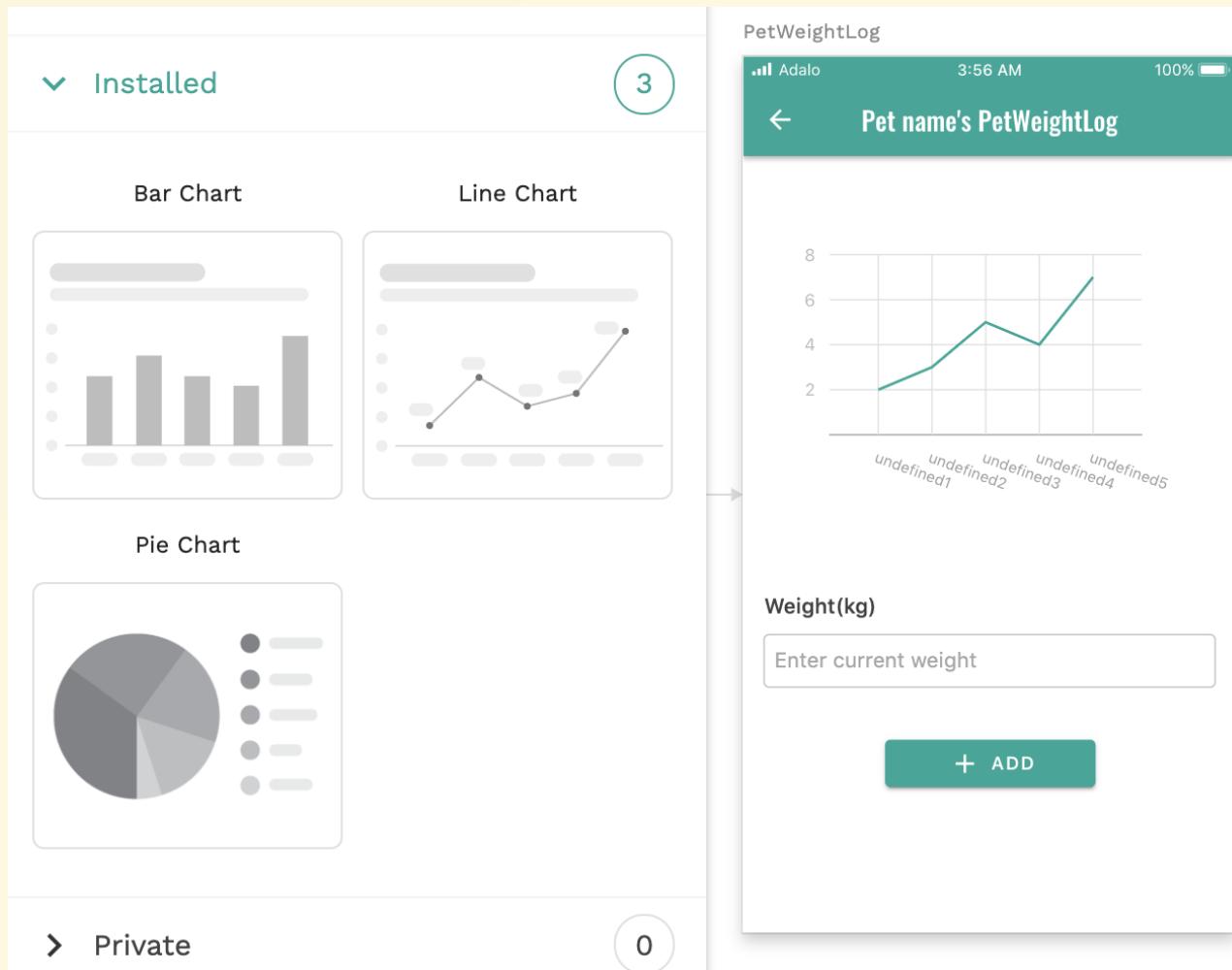
- First, delete the graph that was pasted as an image



- Select "EXPLORE MARKETPLACE" from "ADD COMPONENT"
- Install "Chart Kit"



- Add "Line Chart" to the screen



## Set up a Line Chart.

- Select "PetWeightLogs" in "What is this a chart of?"
- Select "Current Pet" > "PetWeightLogs" in Filter
- In Custom Filter, set "WeightRegisteredTime", "Is after", "30 days ago" to specify display period.
- In Sorting, select "WeightRegisteredTime - Oldest to Newest"

Line Chart  
Line Chart

▼ Line Chart

What is this a chart of?

PetWeightLogs

Filter

Current Pet > PetWeightLogs

Custom Filter

WeightRegisteredTime

Is after

30 days ago

+ ADD ANOTHER FILTER

Sorting

WeightRegisteredTime - Oldest to Ne...

Maximum number of items

No Maximum

ADVANCED OPTIONS

- Set X Axis Value to "PetWeightLog" > "WeightRegisteredTime"
  - Set Date Format to "Date / Time"
- Set Y Axis Value to "PetWeightLog" > "WeightKg"

The screenshot shows a configuration interface for a visualization, likely a line chart. It includes two main panels: one for the X-axis and one for the Y-axis.

**X Axis Value:**

- Label: X Axis Value
- Value: PetWeightLog weightregisteredtime (edit icon)
- Placeholder: T<sup>x</sup>
- Clear button: x<sup>x</sup>

**Y Axis Value:**

- Label: Y Axis Value
- Value: PetWeightLog weightkg (edit icon)
- Placeholder: x<sup>y</sup>

**Point Action:**

- Label: Point Action
- Content: No Actions
- Add Action button: + ADD ACTION

**Styles:**

- Label: Styles

**Right Panel (Configuration):**

- Label: X Axis Value
- Value: PetWeightLog weightregisteredtime (edit icon)
- Placeholder: T<sup>x</sup>
- Clear button: x<sup>x</sup>

**Current PetWeightLog > WeightRegisteredTime:**

- Label: Current PetWeightLog > WeightRegisteredTime
- Icon: trash can icon

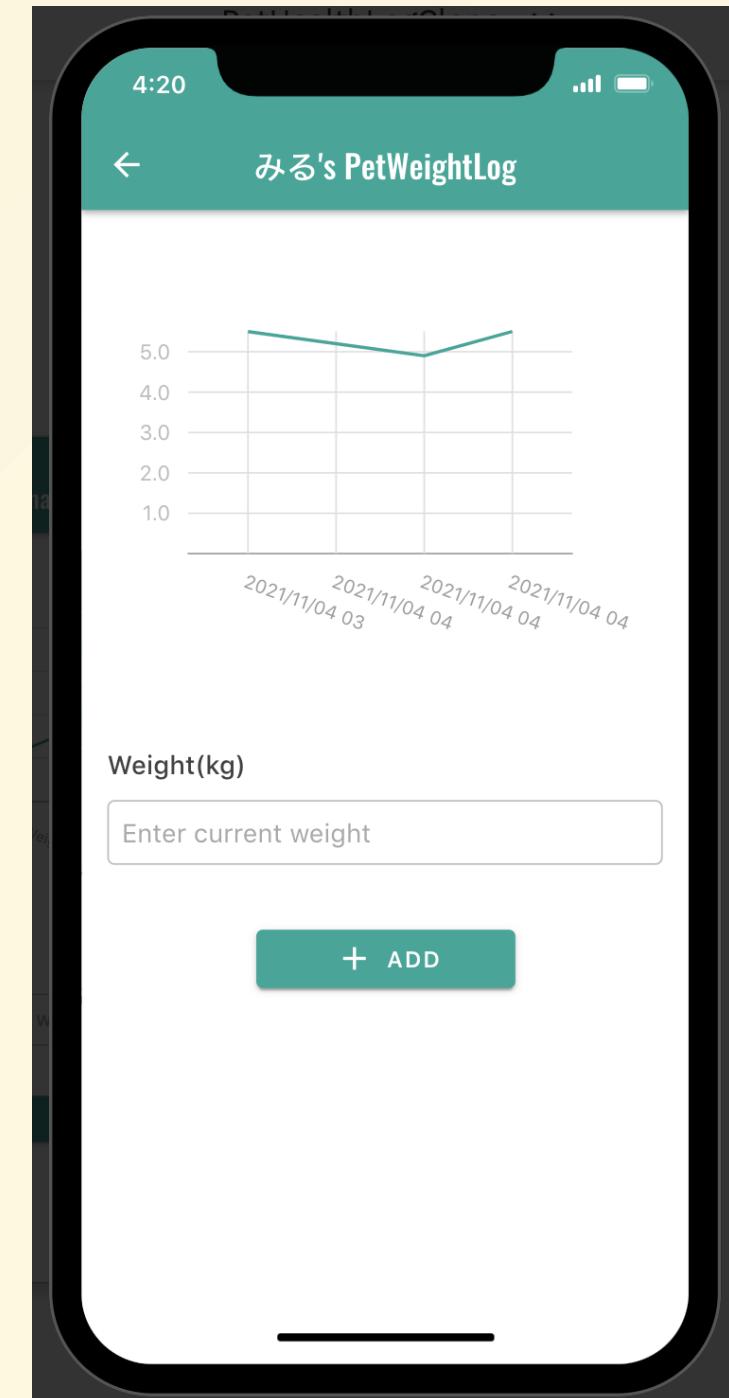
**Date Format:**

- Label: Date Format
- Value: Date / Time
- Icon: dropdown arrow icon

Let's preview the graph. If you add multiple weights, the graph will be drawn.

(The registration time of the weight is too long and will be displayed abbreviated...)

We used the Date&Time type to test it by registering multiple weights on the same day, but it would be better to use the Date type and restrict multiple registration on the same day.)

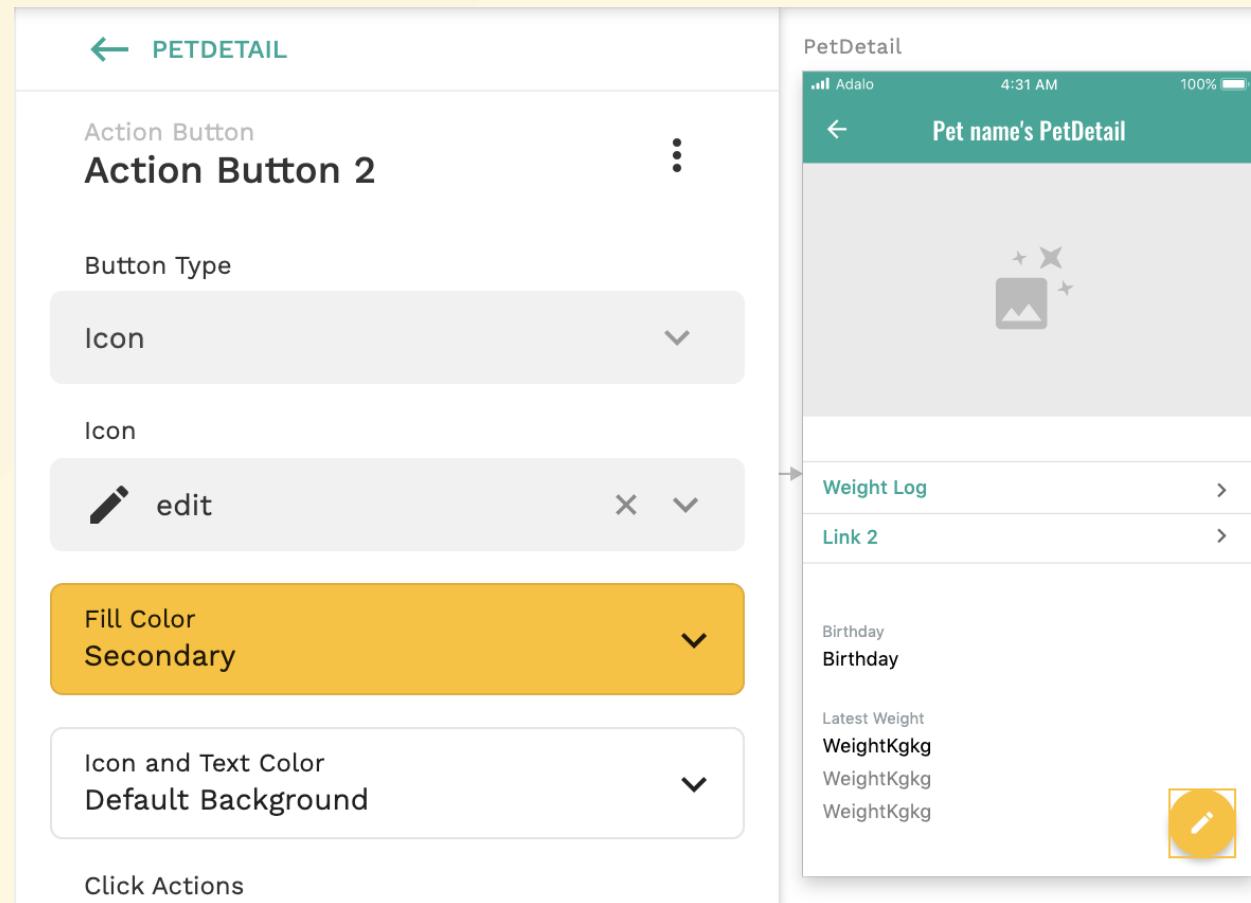


## Updating Data

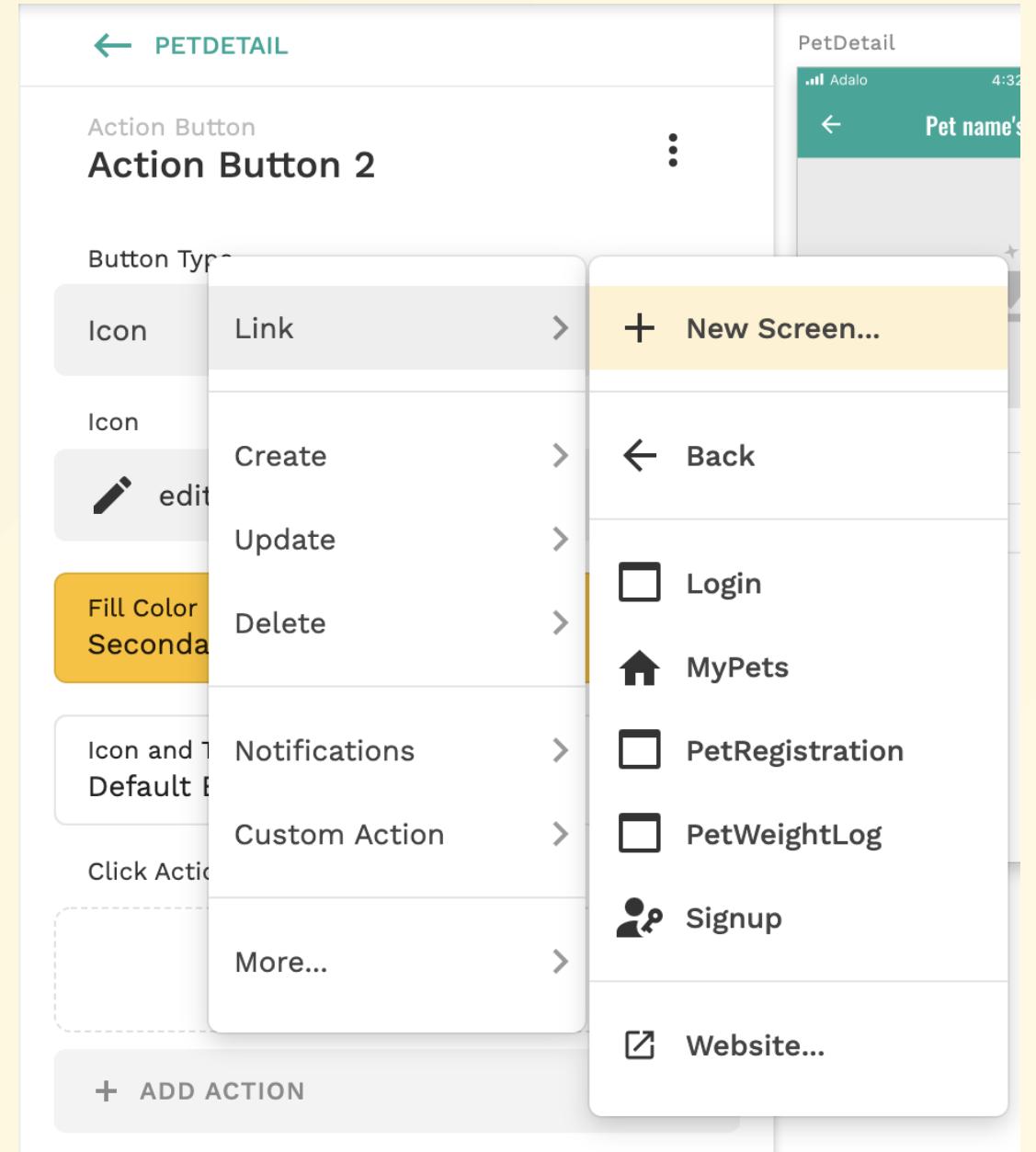
Create a new "EditPetInfo" screen where you can update the information of registered pet.

First, add a link to the Edit Pet Info screen in the Pet Detail screen.

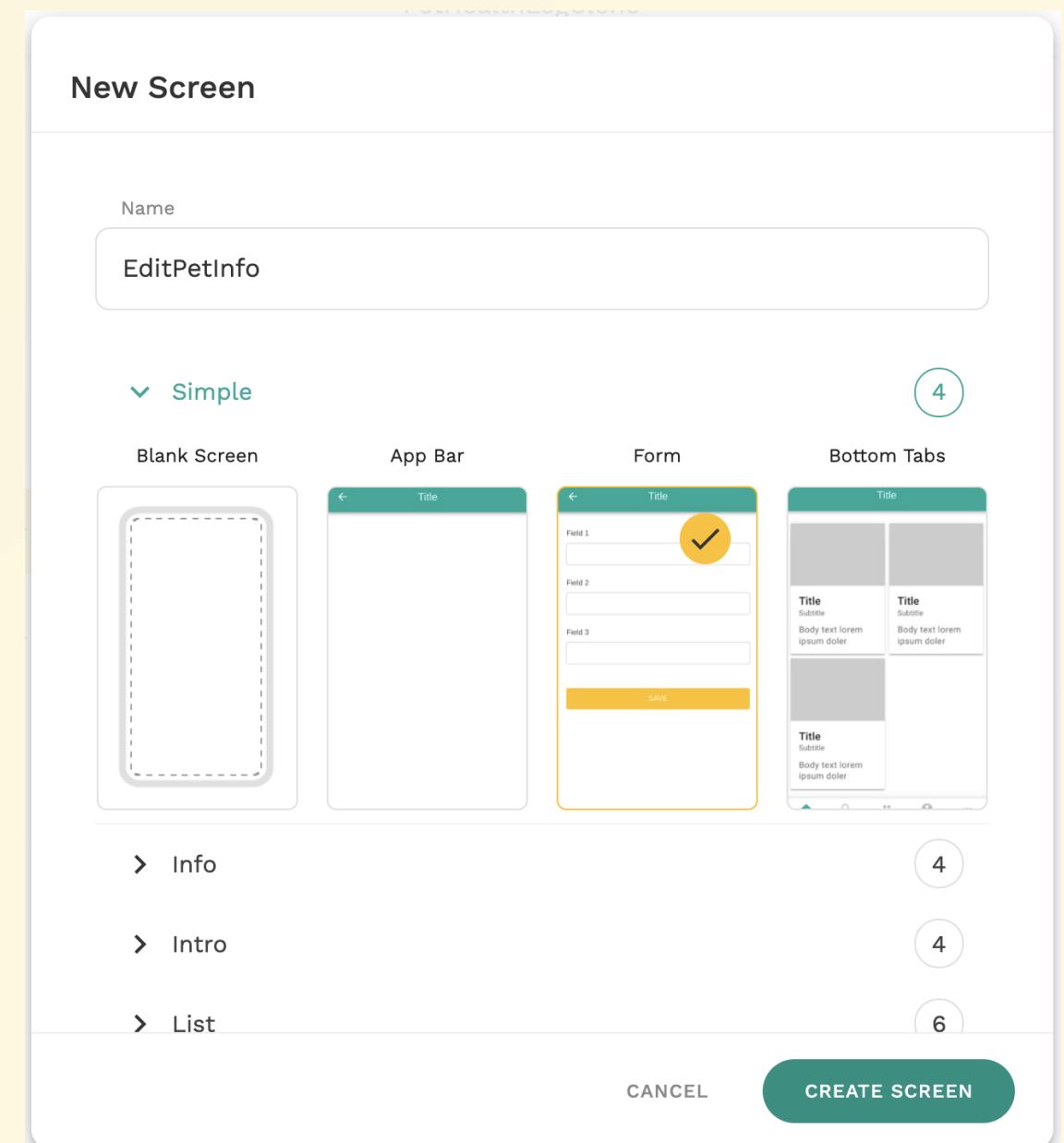
- Add "Action Button" from "ADD COMPONENT"
- Change the Icon to "edit"
- Change Icon and Text Color to "Default Background" (white)



- Select "Link" > "New Screen" from ADD ACTION



- Enter "EditPetInfo" in a Screen Name
- Select "Form"
- Click "CREATE SCREEN"

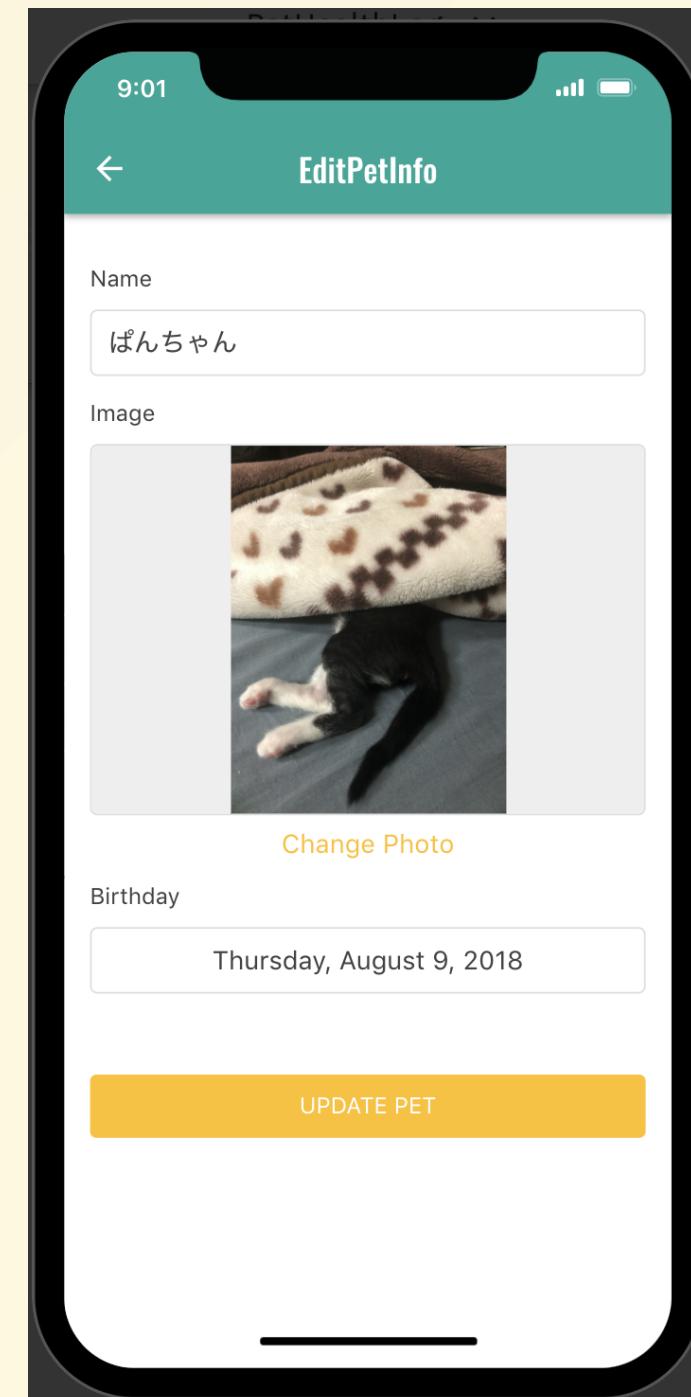


- Select "Pets" in "Which data collection?"
  - The form is automatically generated according to the collection.
- Select "Update Current Pet" in "What do you want the form to do?"
- Change the order of "Birthday" and "Image" in Fields

The image shows the Adalo platform interface for creating a mobile form. On the left, the 'EDITPETINFO' screen is displayed with the title 'Form' and a dropdown menu set to 'Pets'. Below it, another dropdown menu is set to 'Update Current Pet'. On the right, the resulting mobile screen 'EditPetInfo' is shown on an iPhone with the status bar 'Adalo 4:37 AM 100%'. The mobile screen has three fields: 'Name' (placeholder 'Enter name...'), 'Birthday' (placeholder 'Thursday, May 4th, 2019'), and 'Image' (with a 'Choose Photo' button). A large orange 'UPDATE PET' button is at the bottom. Arrows indicate the flow from the configuration screen to the preview screen.

EditPetInfo screen is ready.

Make sure that you can edit the pet information.



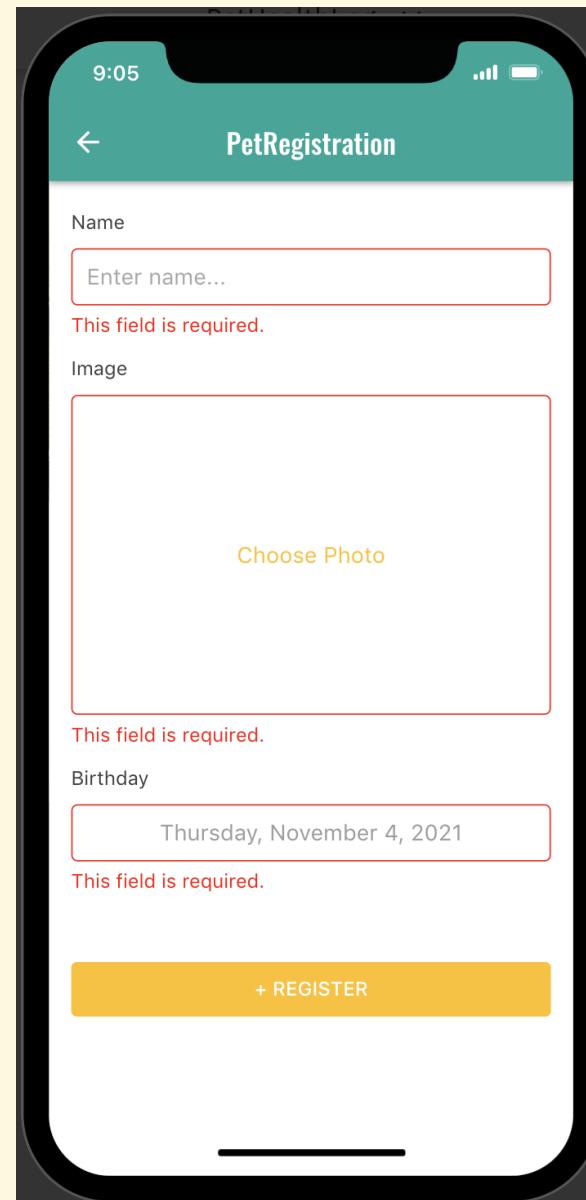
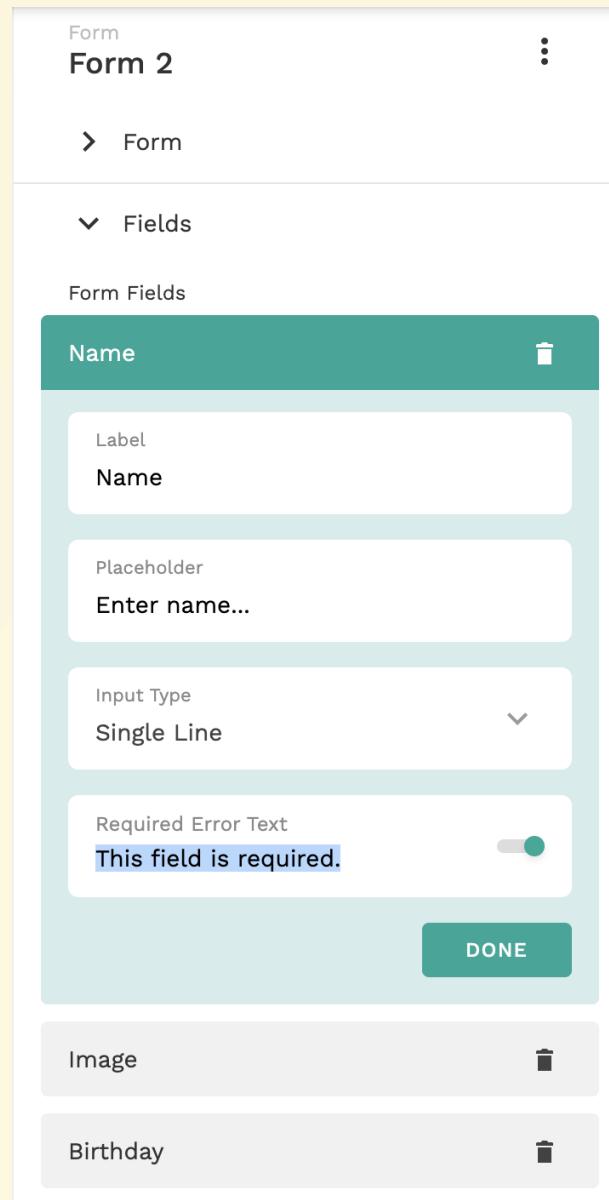
## Supplement

- In the previous article, we created the input form for the pet registration screen by adding components one by one, but it can be automatically generated in the same way as we created the pet edit screen.

- Let's delete all components except "AppBar" in the pet registration screen, add a "Form" component, and specify "Pet Collection" to generate the form automatically.
  - Then, manually add link to the pet list screen after registration by adding action to the Submit Button.

# Additional Information

- Using the Form component, you can check if the required fields are filled in by just checking "the Required Error Text" of the field item.
- Use the Form component as much as possible when creating input forms.



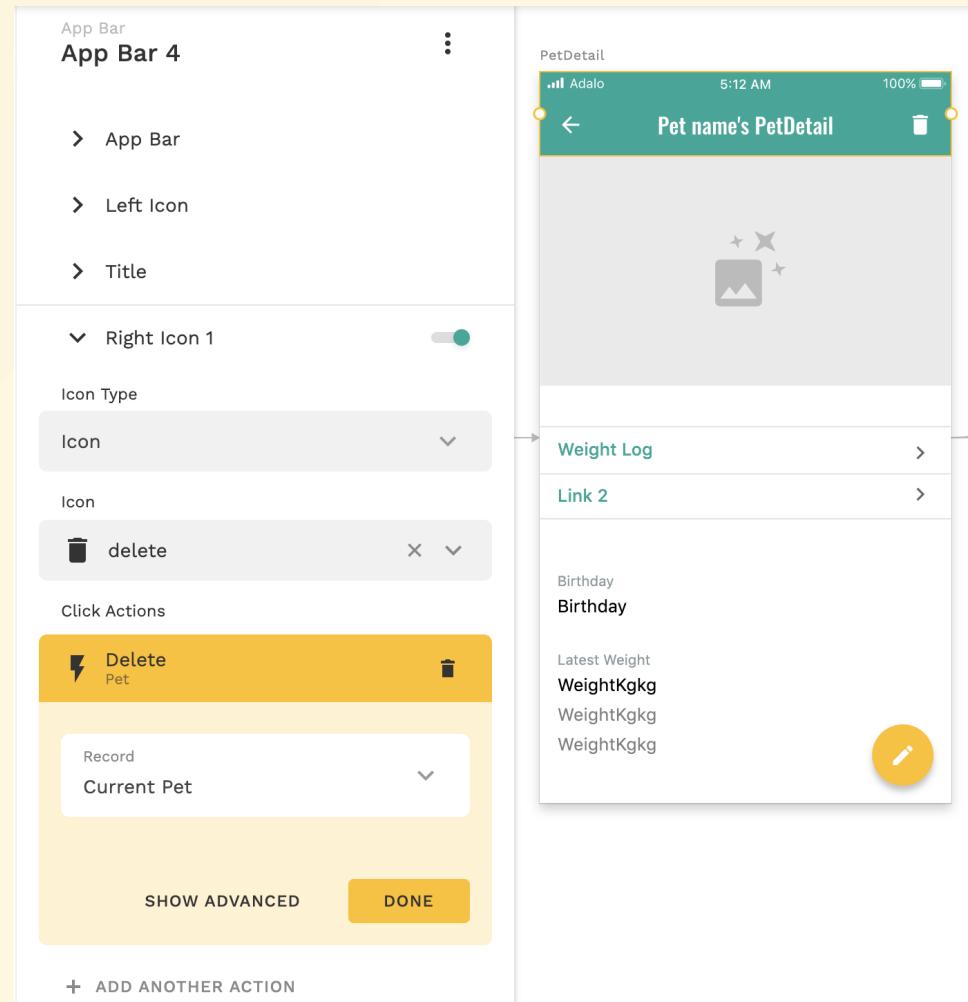
## Reference: The actual flow of application development in Adalo

- In the sample application, we have proceeded with the following flow: UI creation > database design > linkage between them.
  - This is because it seems difficult to design the database without knowing what kind of screen and data we need.
- When actually developing an application with Adalo, It is recommended that first draw a UI sketch and identify the necessary data, and then design the database. By doing so, you can take advantage of automatic generation for UI creation.
  - However, the actual development would not be a one-way flow, but a trial-and-error process, alternating between database design and UI design.

## Deleting Data

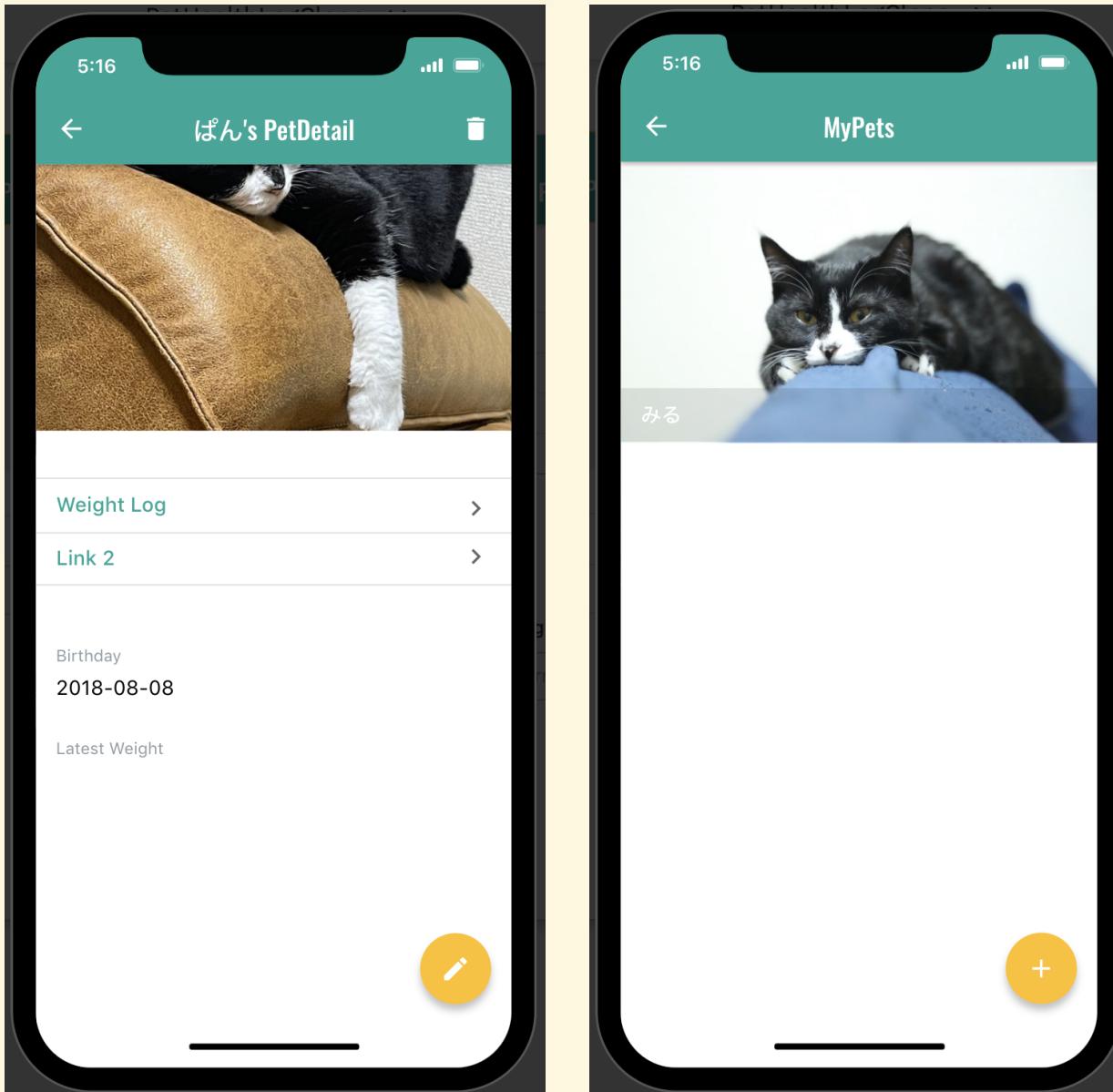
Create a button to delete a registered pet in the pet details screen.

- Select the App Bar and turn on the "Right Icon 1"'s toggle
- Change Icon to "delete"
- Select "Delete" > "Current Pet" from "ADD ACTION"
- Select "Link" > "Mypets" from "ADD ANOTHER ACTION"



Let's try deleting with the Preview function.

When the deletion is completed, you will be redirected to the Pet List screen. The deleted pets will not be displayed.



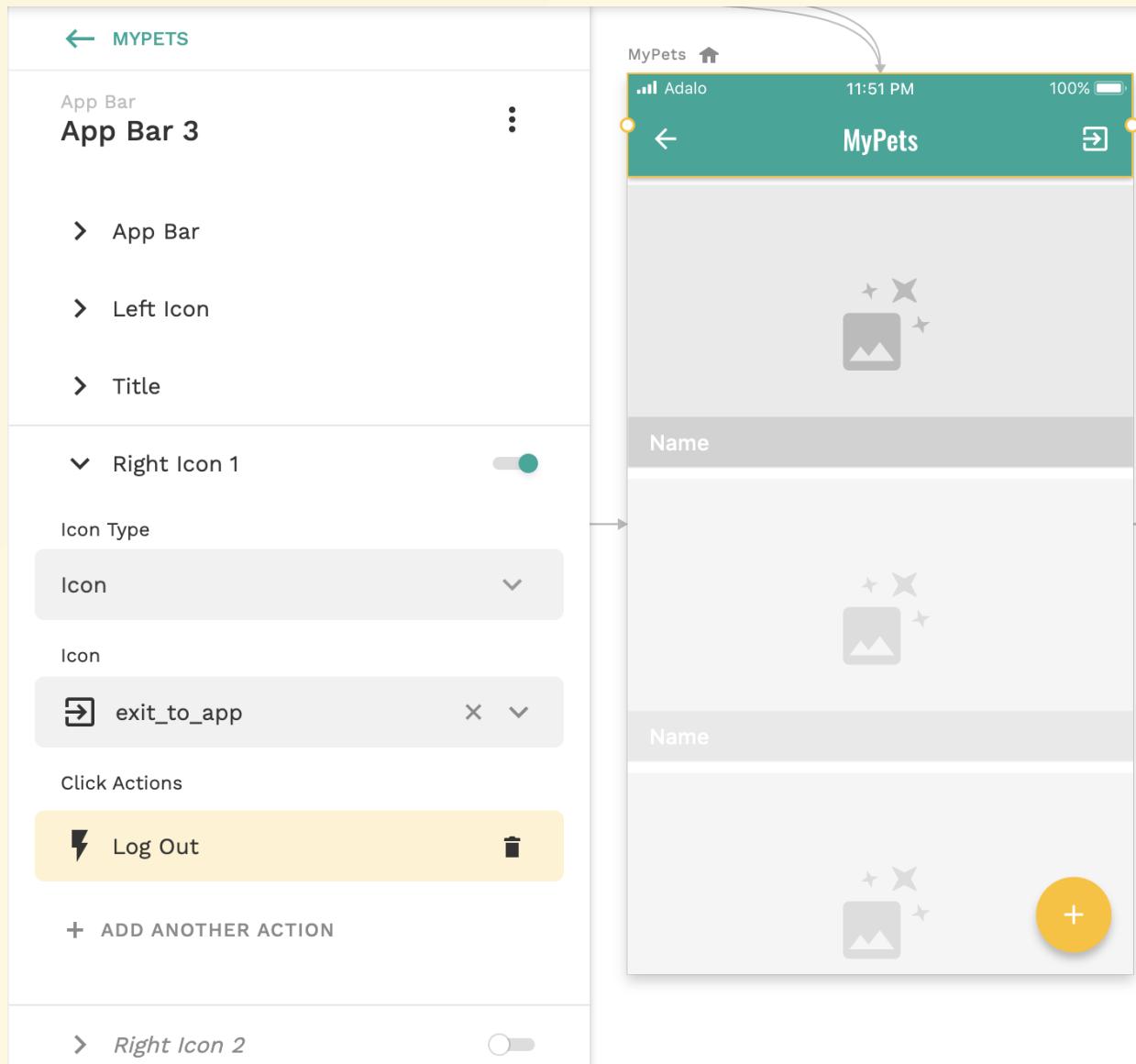
We have now implemented all kinds of CRUD(CREATE, READ, UPDATE, DELETE) operations.

# Improve sample application

Let's improve the sample application by using the features of Adalo that we haven't learned yet.

# Logout

- Select the AppBar in the pet list screen, and activate "Right Icon 1"
- Change Icon to "exit\_to\_app"
- Select "ADD ACTION" > "More..." > "User Login" > "Log Out"
- Select "ADD ANOTHER ACTION" > "Link" > "Login"

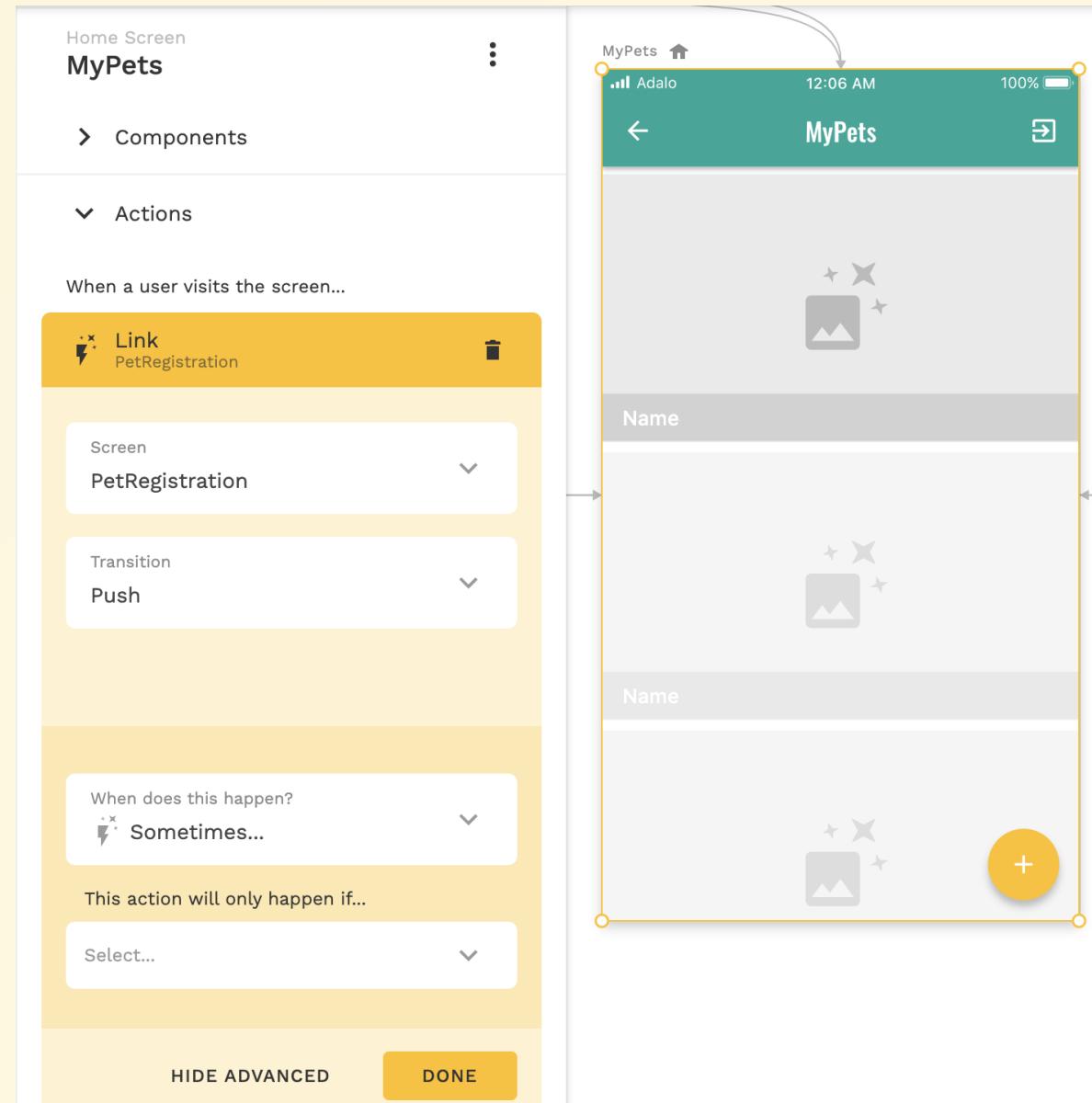


Check it with the Preview function.

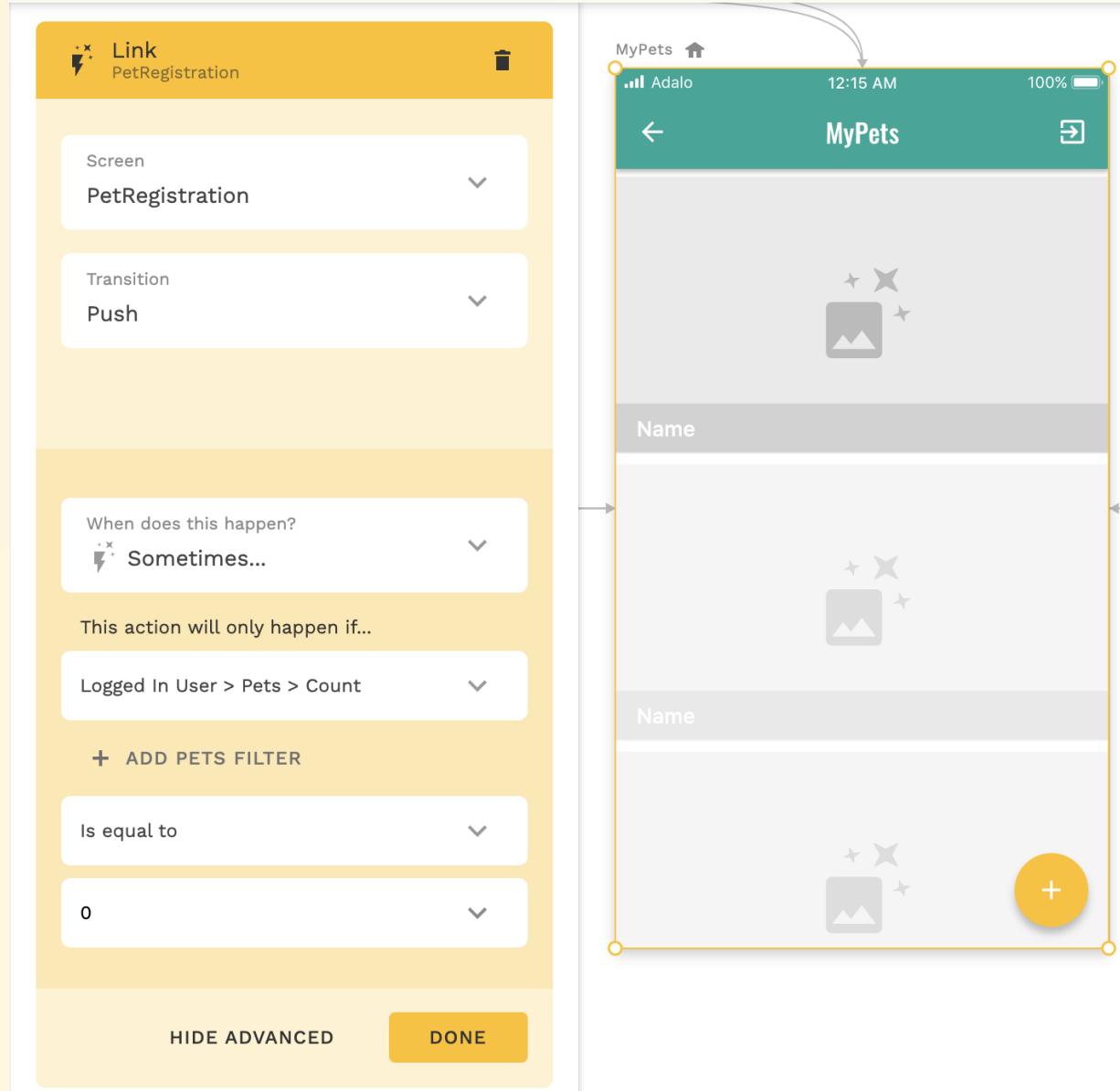
# Action execution condition setting

If no pets have been registered, redirect users from Pet List screen to Pet Registration screen.

- In the Pet List screen, select "Actions" > "ADD ACTION" > "Link" > "Pet Registration"
- Click "SHOW ADVANCED" and change "When does



- Set "This action will only happen if..." "More" > "Logged In User's" > "Pets" > "Count"
- Change the number under "Is equal to" to 0.



Check it out with the Preview function.

When you sign up as a new user, you will be redirected from the Pet List screen to the Pet Registration screen.

## Selective Input Form

Add gender to the pet's information and allow selective input in the input form.

The choices used in the selective input form are prepared as records by adding a collection to the database.

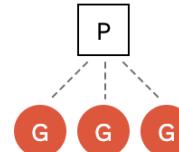
- Add a Genders Collection to the database (leave the Property as default)
- Click "0 Records" > "ADD GENDER" and add two Records, "Male" and "Female".

Genders				<a href="#">+ ADD GENDER</a>				
	Name	Created	Updated					
<input type="checkbox"/>	Female	8 minutes ago	8 minutes ago					
<input type="checkbox"/>	Male	8 minutes ago	8 minutes ago					

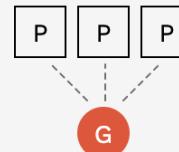
- Add a one-to-many Relationship to the Genders Collection with the Pets Collection
  - Since one pet has one gender, and one gender is set for multiple pets

How are Pets & Genders related?

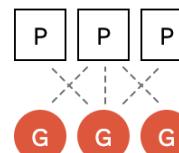
P = Pets    G = Genders



A Pet can have multiple Genders  
A Gender belongs to one Pet



A Pet can only have one Gender  
A Gender can have multiple Pets

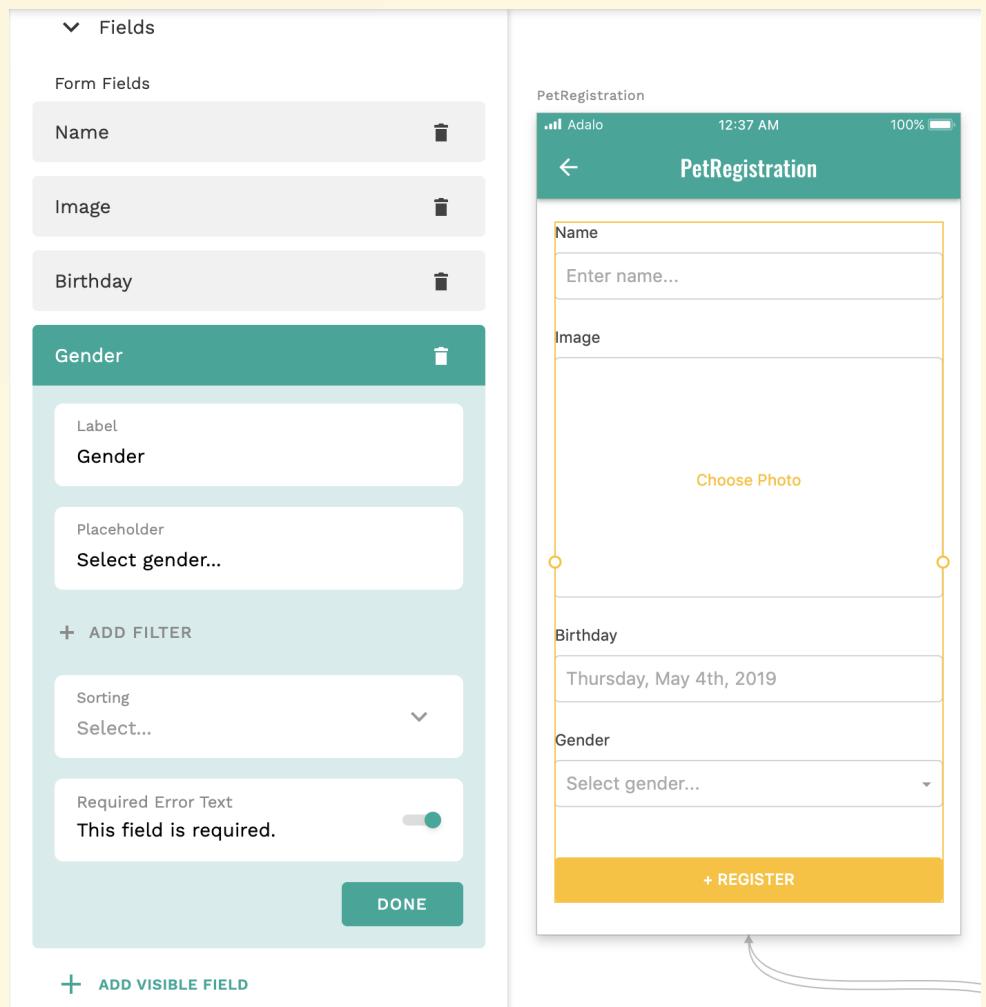


A Gender can have multiple Pets  
A Pet can have multiple Genders

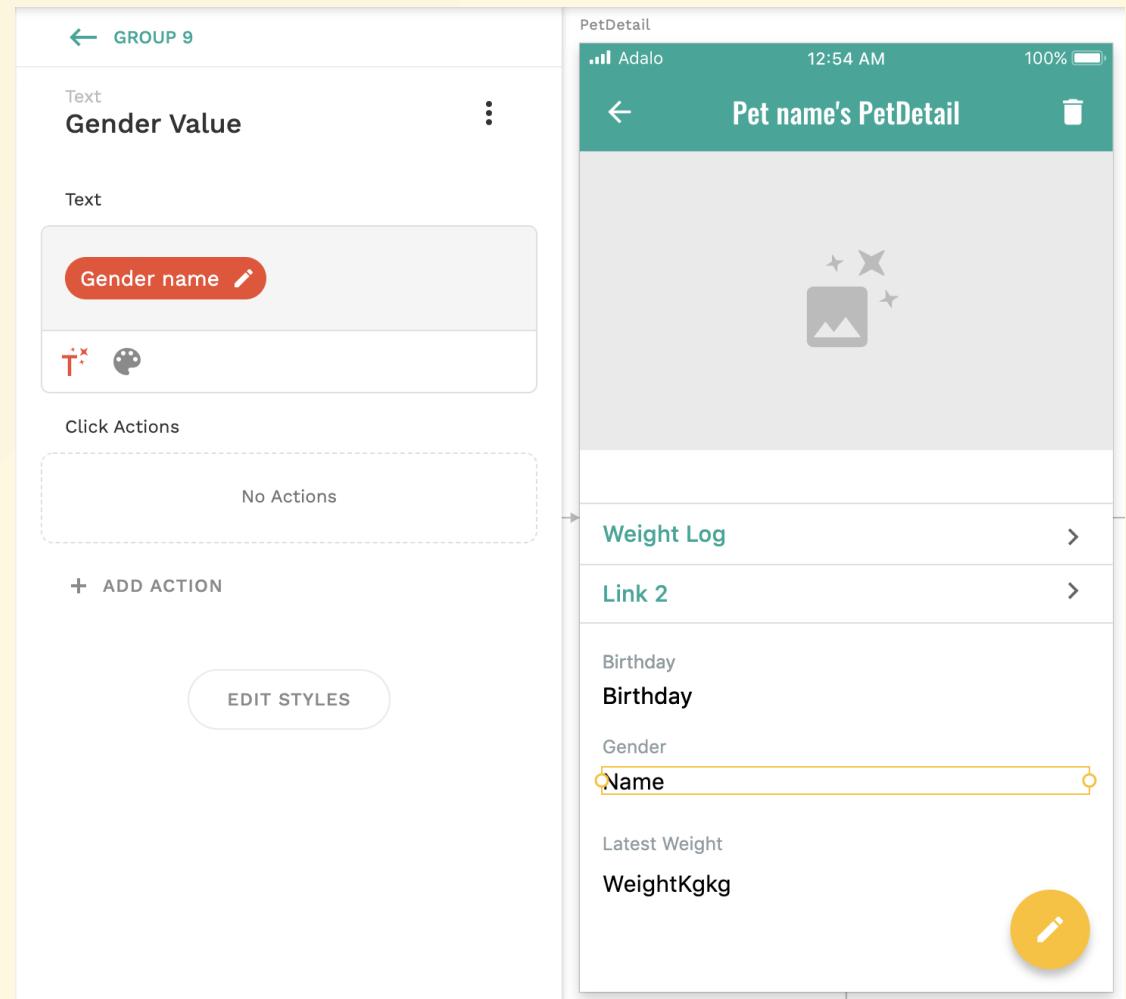
CANCEL

DONE

- Select the form on the pet registration screen
- Select "Fields" > "ADD VISIBLE FIELD" > "Gender"

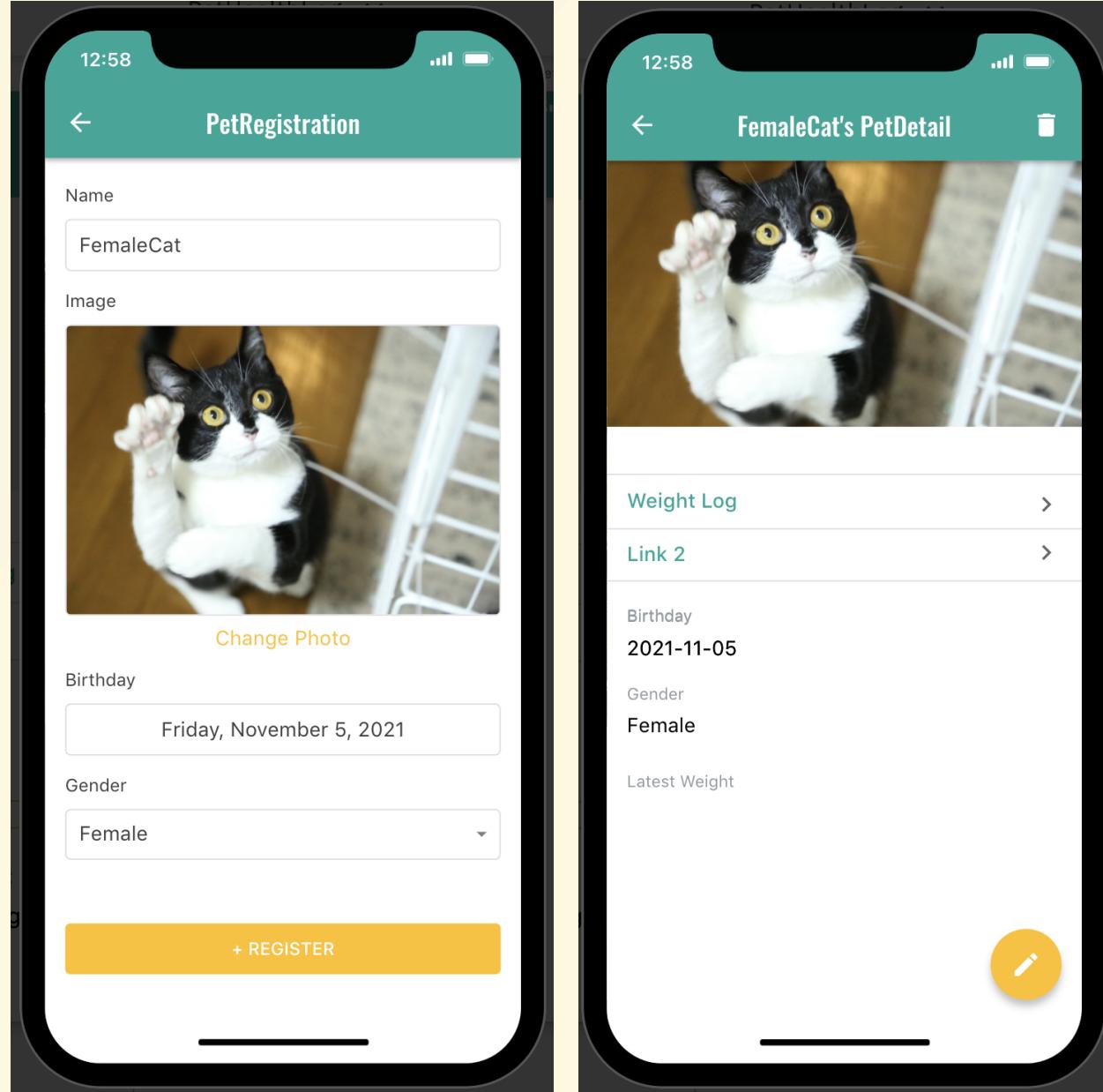


- Add a Gender field to the Pet Detail screen.



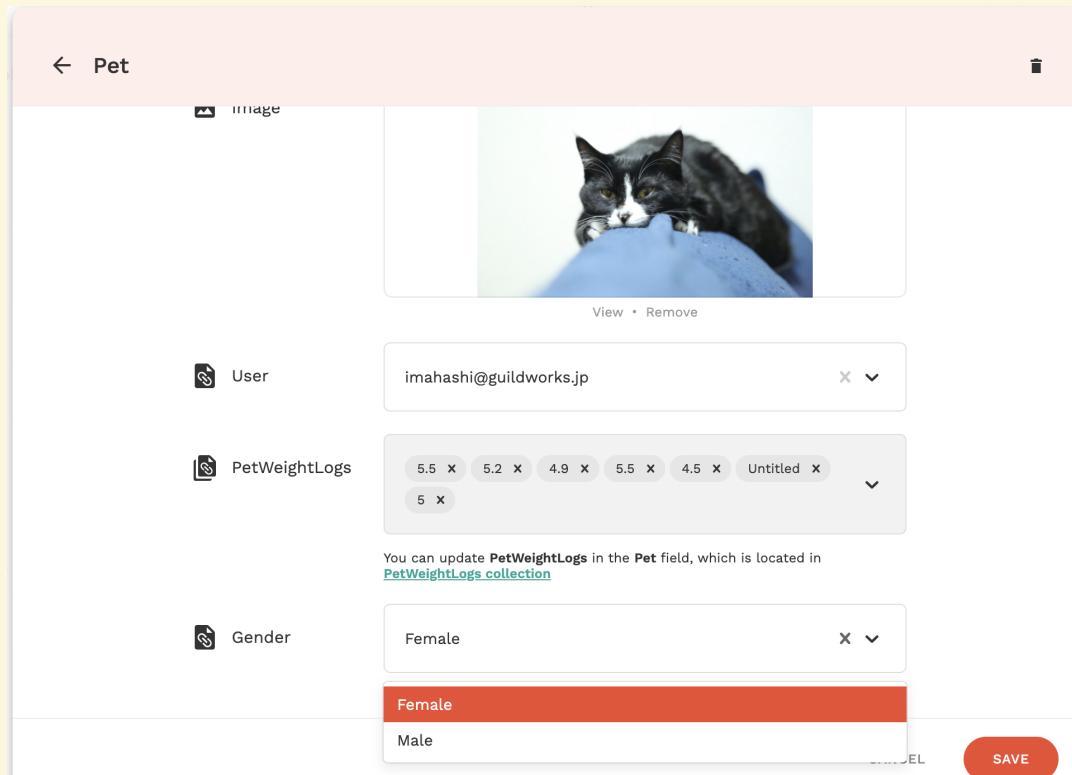
Check it out in the Preview function.

- You can select the gender on the pet registration screen.
- The selected gender will be displayed on the Pet Details screen.



# Supplement

- If you are bothered by the empty fields in the pet you registered before you were able to select the gender, you can go to the Pet Collection Record, click on the pet, and set the gender manually.



## Reference

You can create a multiple-choice input form using Marketplace's MultiselectDropdown.

If you need it, try it out.

 Marketplace



Multiselect Dropdown

Select multiple items from a dynamically-generated list.



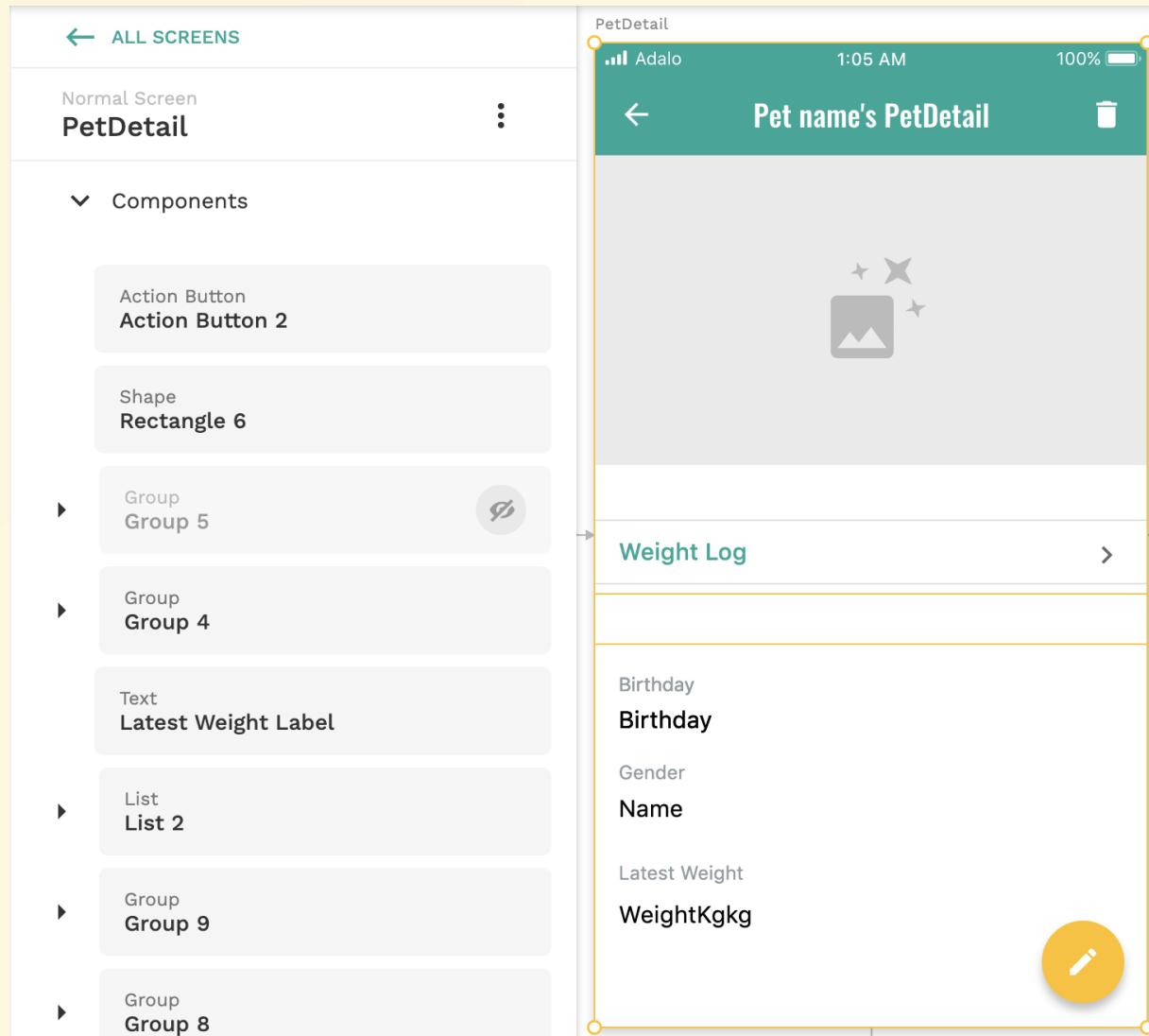
By Conroy Whitney 

INSTALLED

# Hiding components

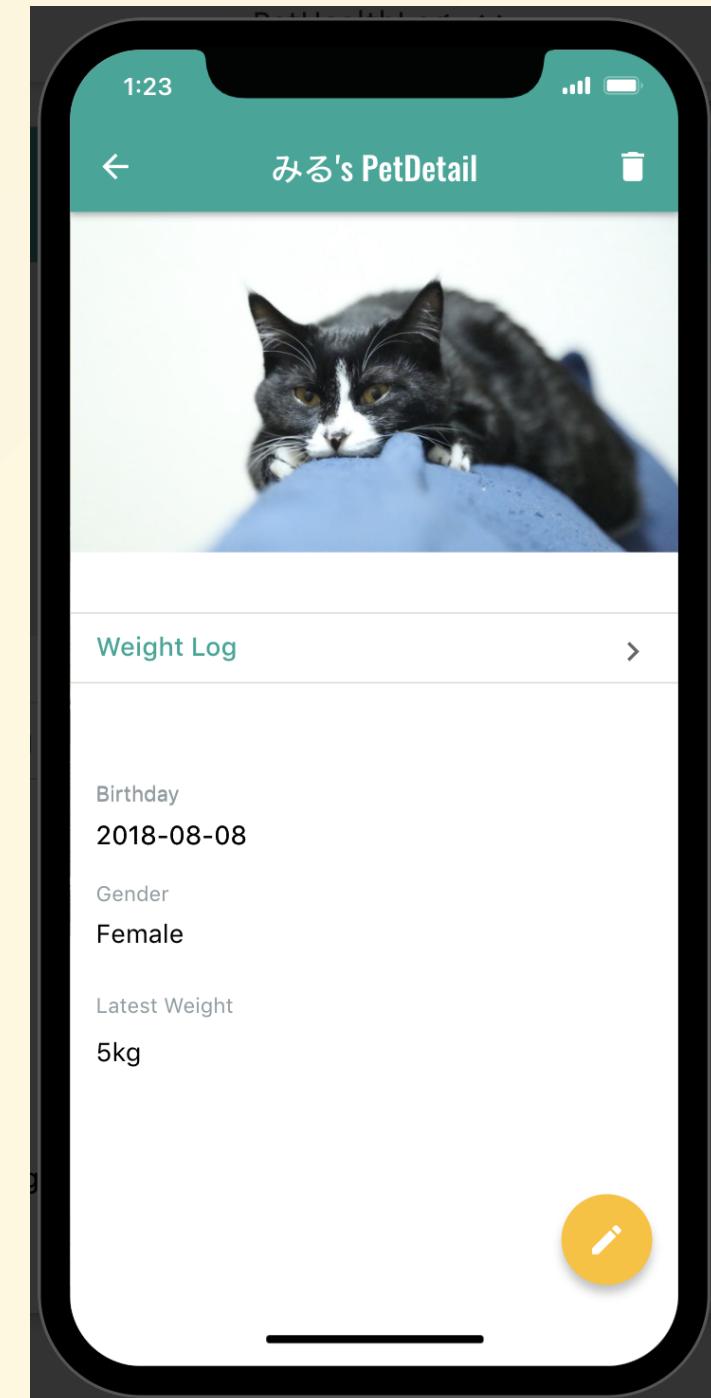
You can keep the components which you do not want to display without deleting them.

- In the component list of the Pet Details screen, mouse over the Group that contains "Link 2", and click the eye icon on the right side



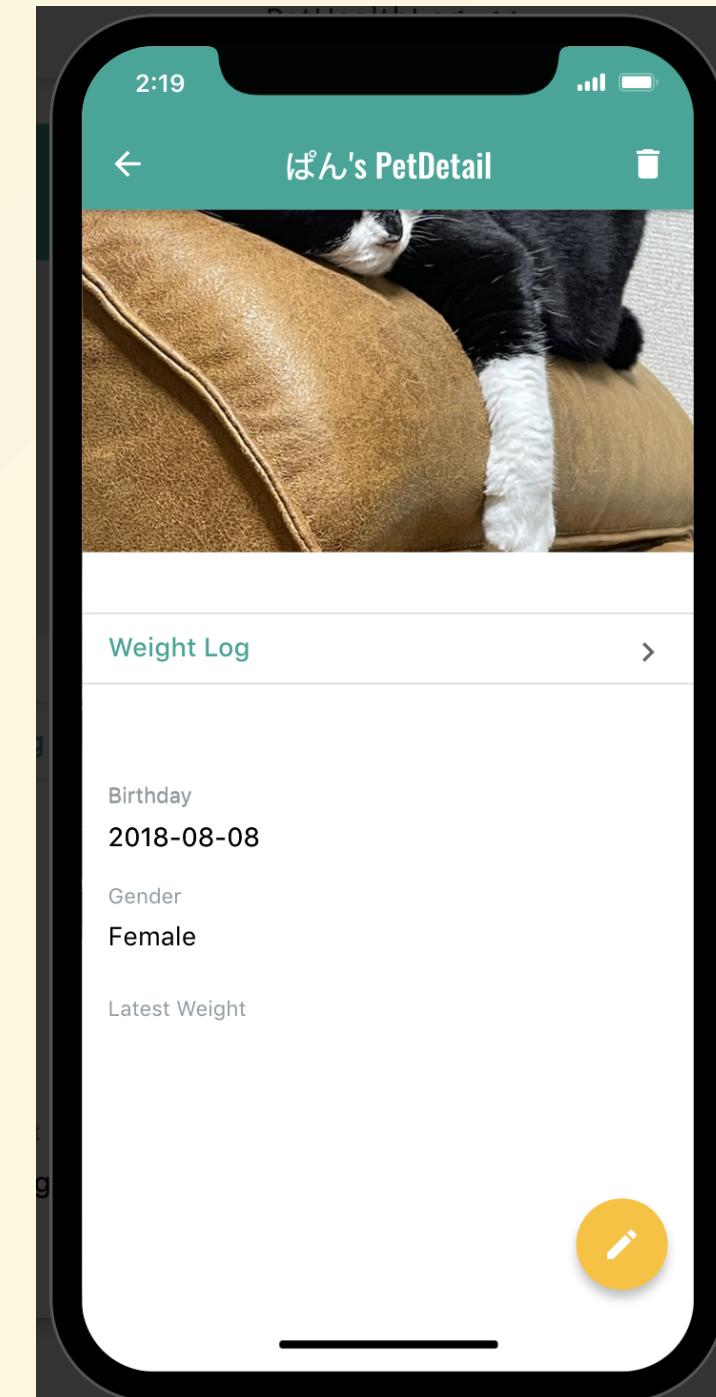
If you check the Preview function, you will see that "Link 2" has disappeared.

You can make it appear again by clicking on the icon a second time.

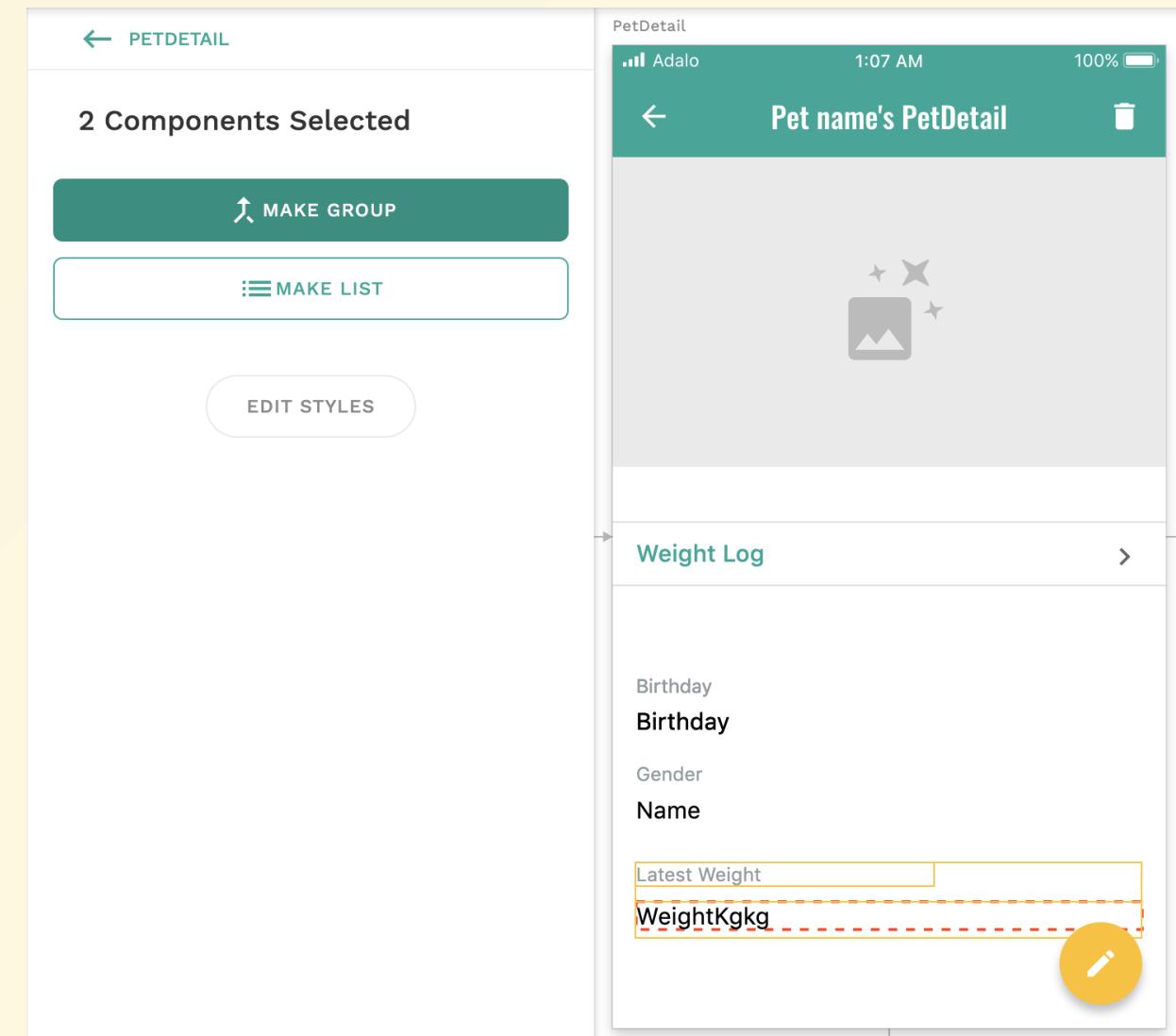


# Show or hide components depending on conditions

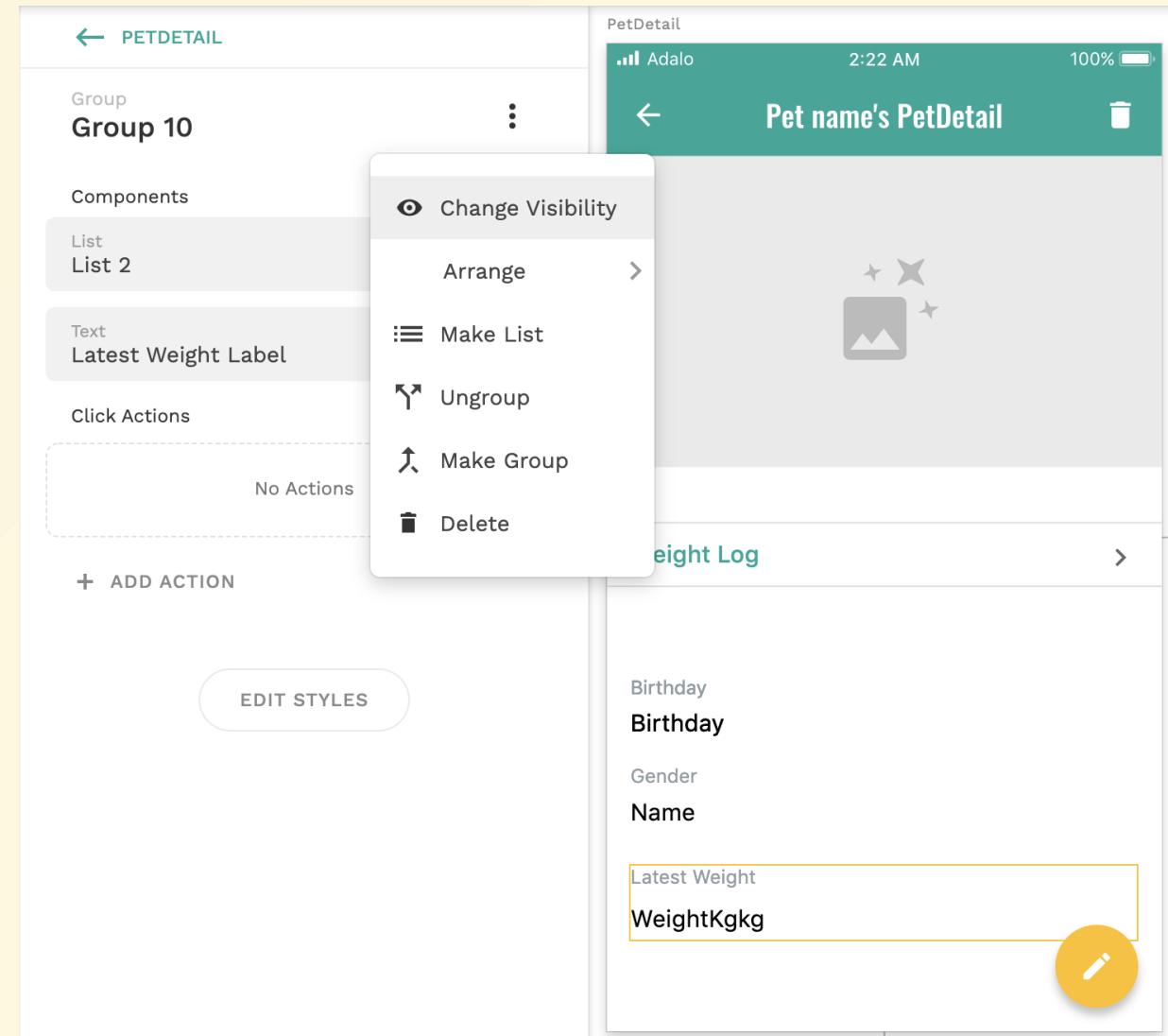
If the weight is not registered, the Latest Weight in the Pet Details screen should be hidden. Let's configure this.



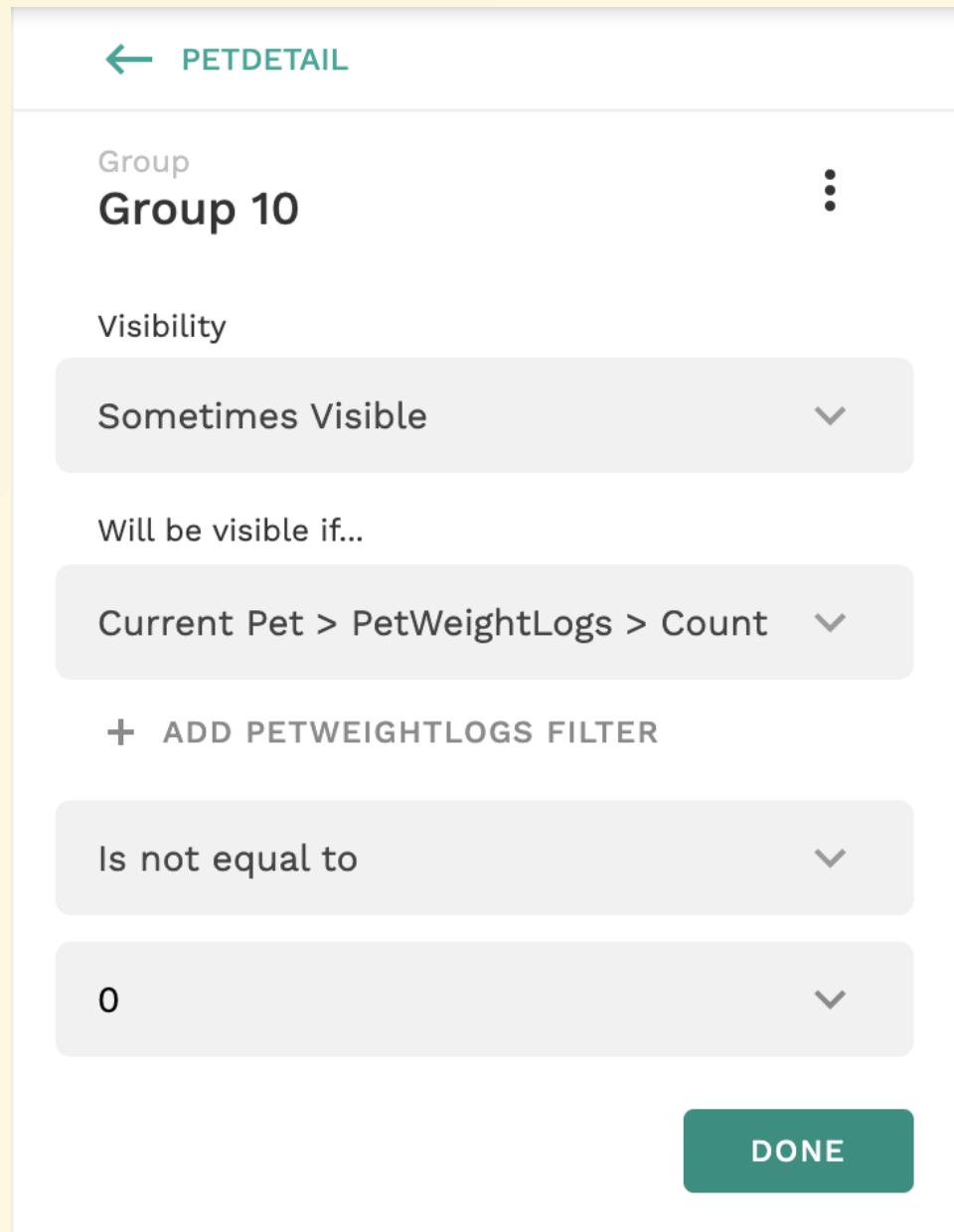
- Select the label "Latest Weight" and its value, and group them together



- Select Change Visibility

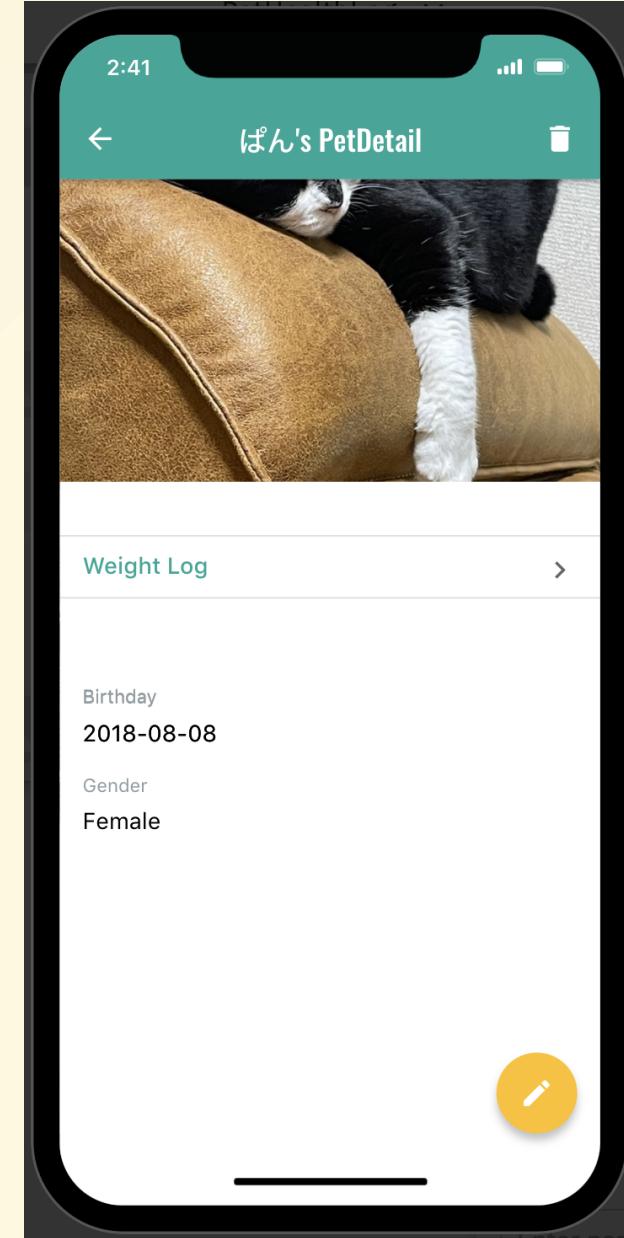


- Change Visibility to "Sometimes Visible"
- Set "Will be visible if..." as "Current Pet" > "PetWeightLogs" > "Count"
- Set "Is not equal to 0"



Checking with the Preview function, if the weight is unregistered, the Latest Weight is now hidden.

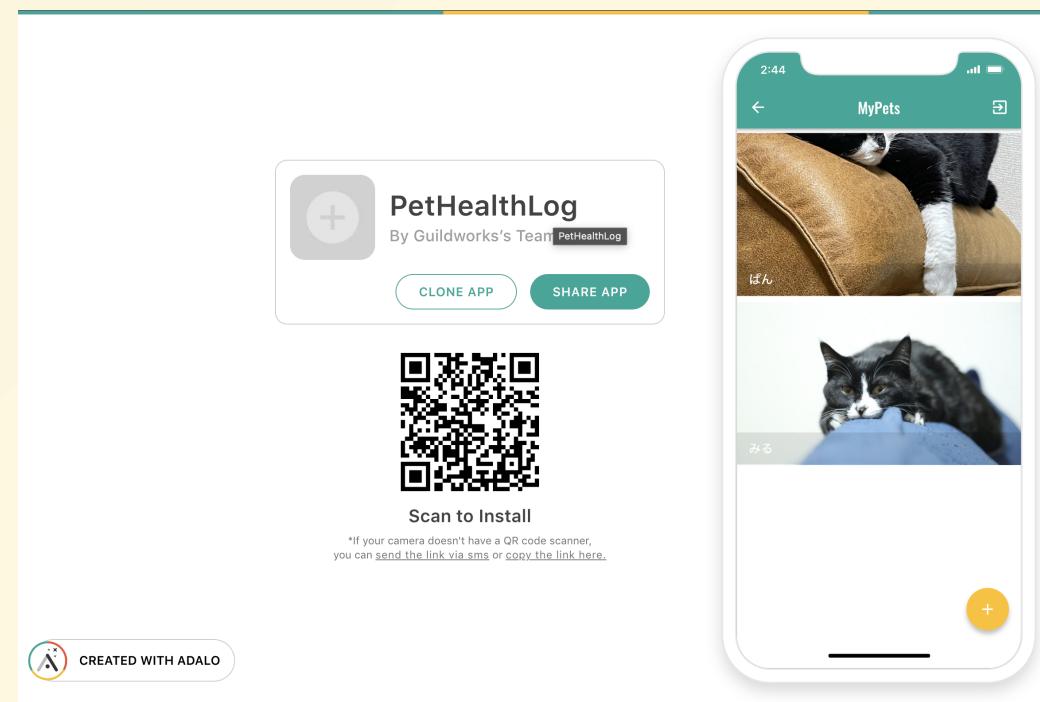
This is the end of the improvement of the sample application.



# URL for cloning

- You can use the following URL to clone the application that reflects the work done so far, and use it to check your answers.

<https://previewer.adalo.com/f1324ea8-ec47-4c22-a3a9-3258044eb754>



# External Integration

If there is something you cannot achieve with Adalo alone, you may be able to achieve it by integrating your adalo app with external services.

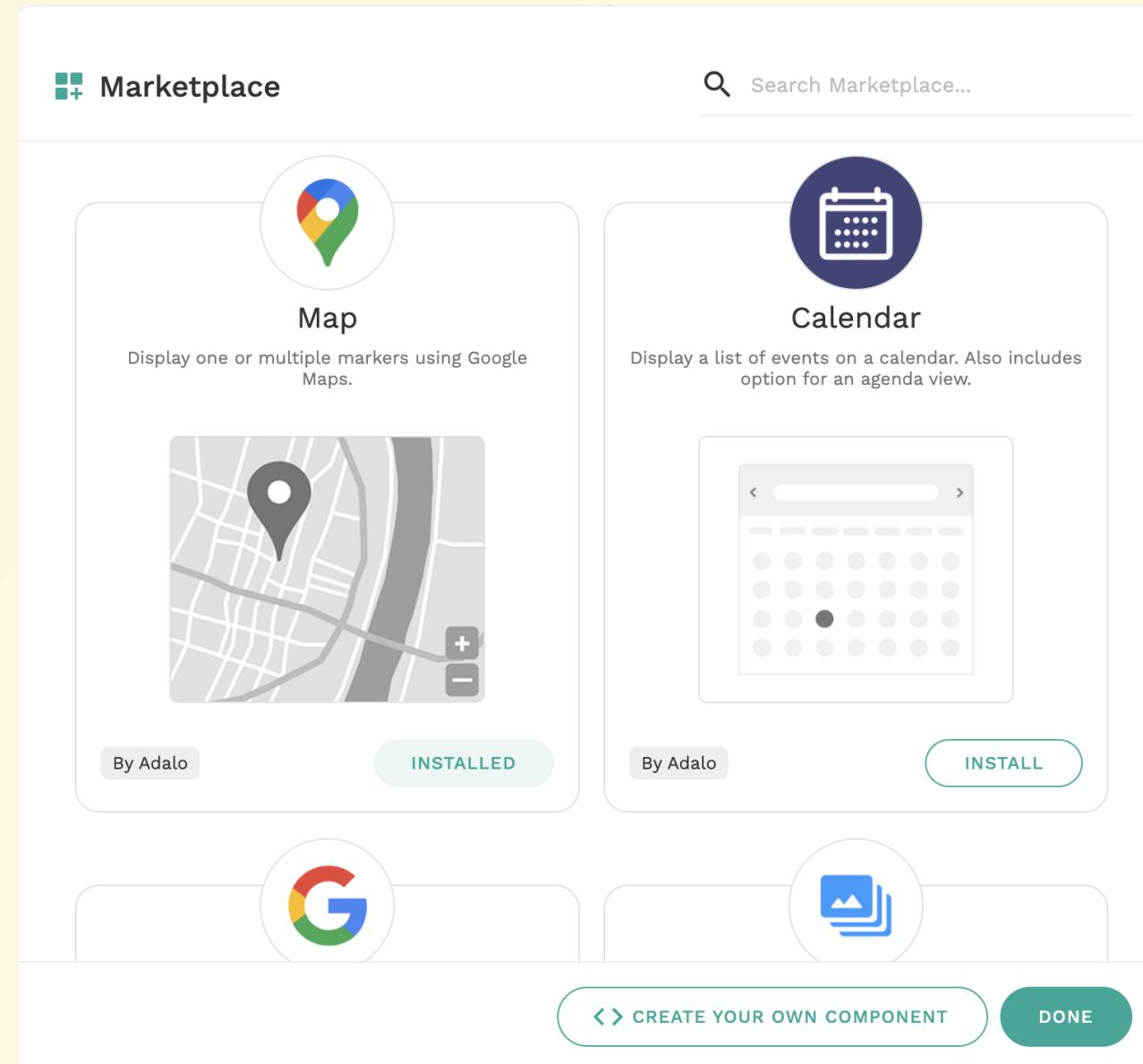
I will show you how to do that.

Let's learn following four methods of external integration.

- Marketplace external integration component
- Custom Action
- External Collection
- Connector service

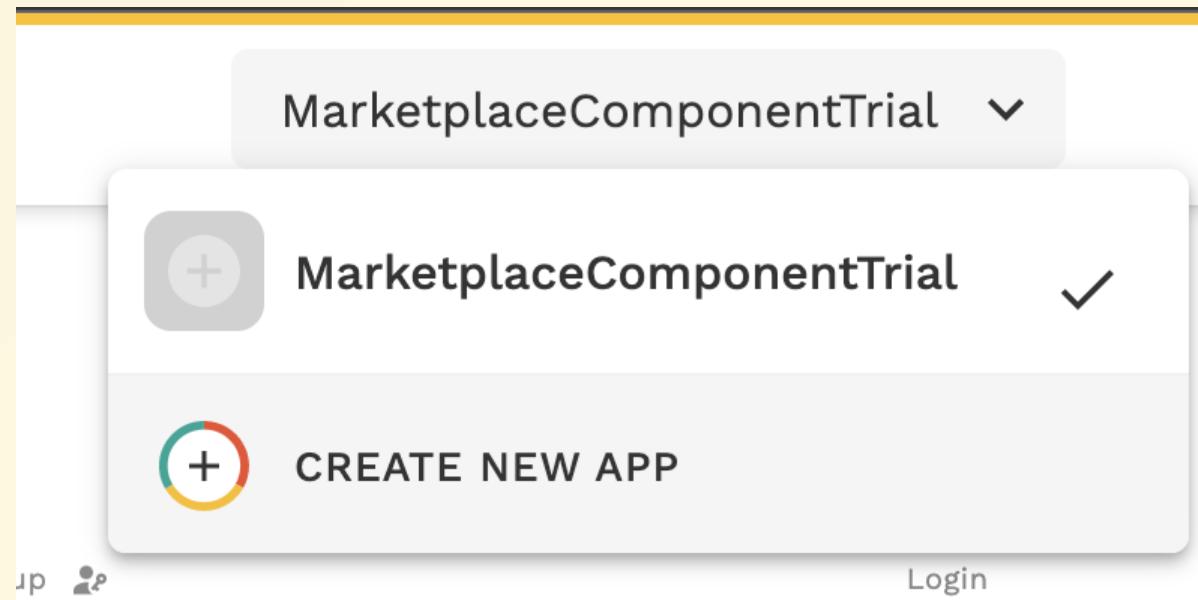
# Marketplace external integration components

You can add a component that enables external integration from Marketplace.

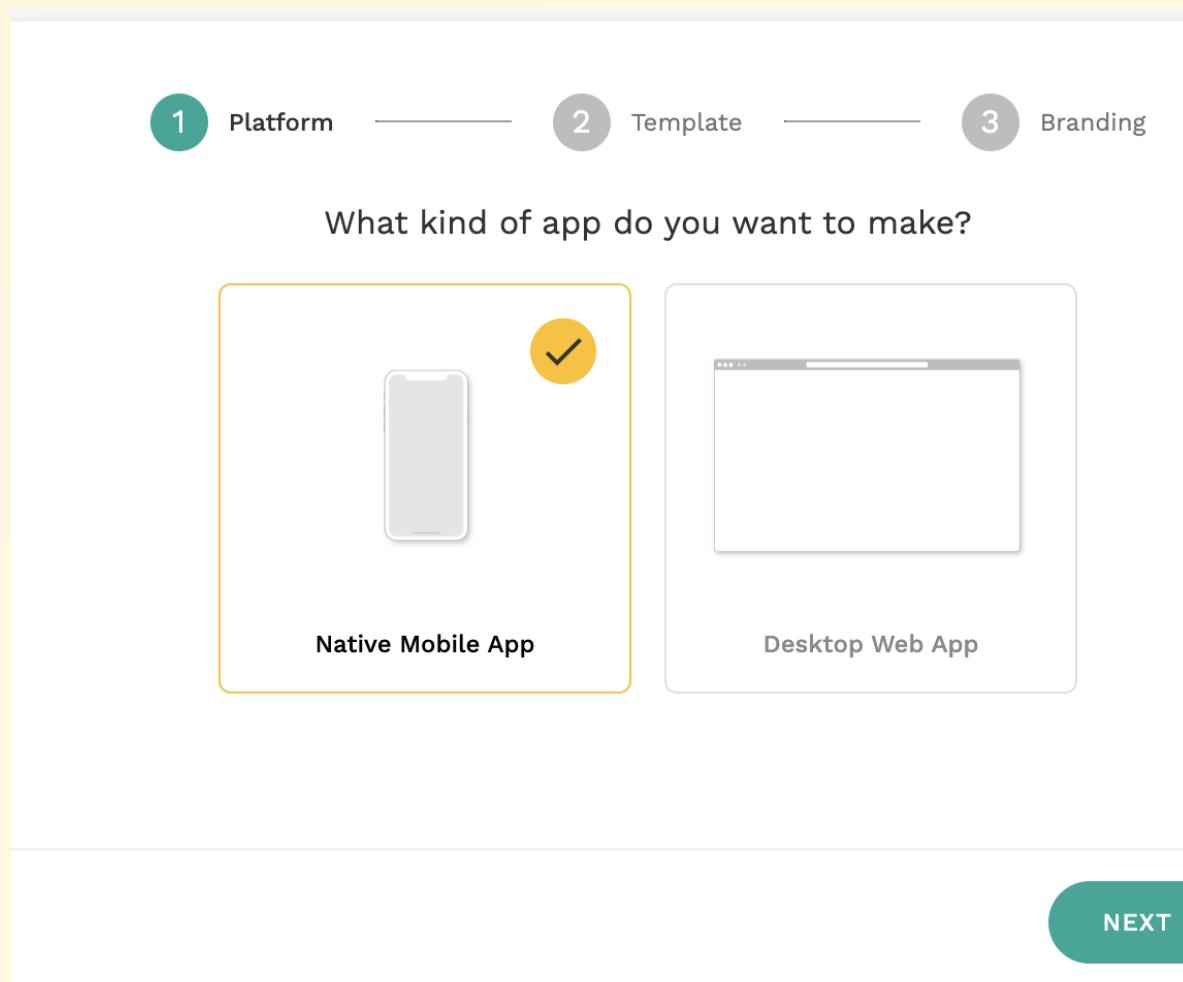


First, let's create a new application.

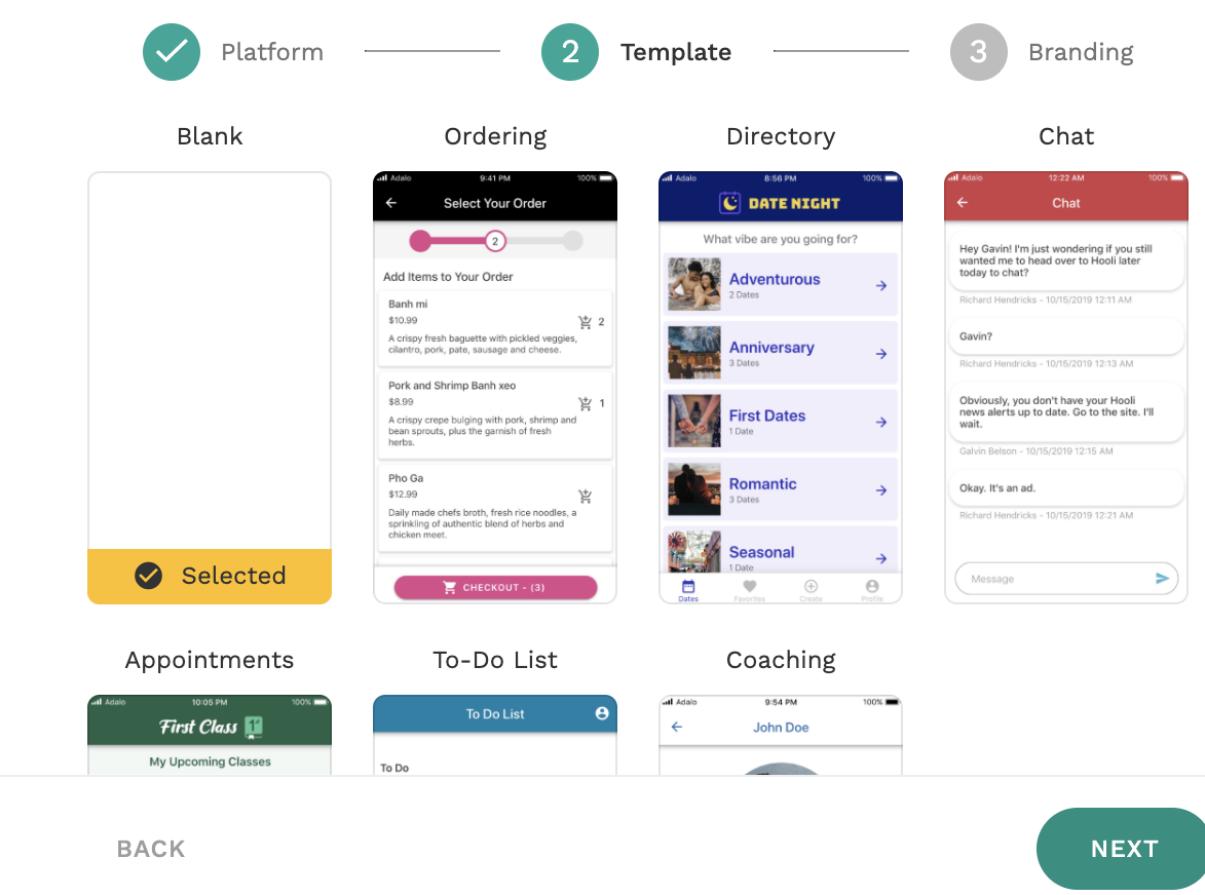
- Login to Adalo  
Go to  
<https://app.adalo.com/login>
- Click CREATE NEW APP



- Select Native Mobile App for Platform
- Click "Next"



- Select Blank for Template.



- For App Name, enter "MarketplaceComponentTrial".
- Please set the Color freely.

The screenshot shows a branding configuration screen. At the top, there are three tabs: 'Platform' (with a checkmark), 'Template' (with a checkmark), and 'Branding' (highlighted with a green circle and the number '3'). Below the tabs, the 'App Name' field contains 'MarketplaceComponentTrial'. Under the 'Branding' section, there are two dropdown menus: 'Primary Color' set to '#00A898' (teal) and 'Secondary Color' set to '#FFC00E' (yellow). At the bottom, there are 'BACK', 'ADVANCED OPTIONS', and 'CREATE' buttons.

Platform

Template

3 Branding

App Name

MarketplaceComponentTrial

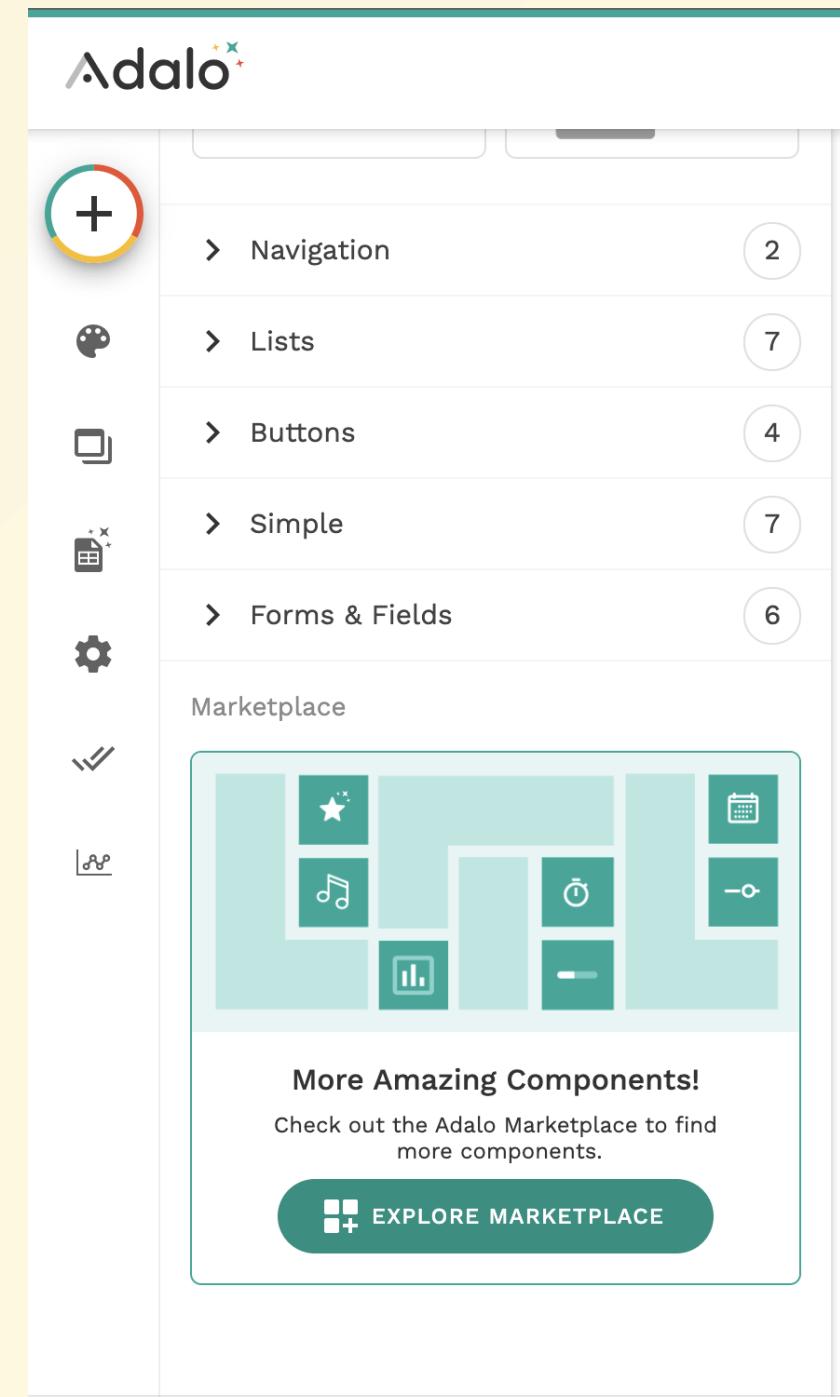
Primary Color  
#00A898

Secondary Color  
#FFC00E

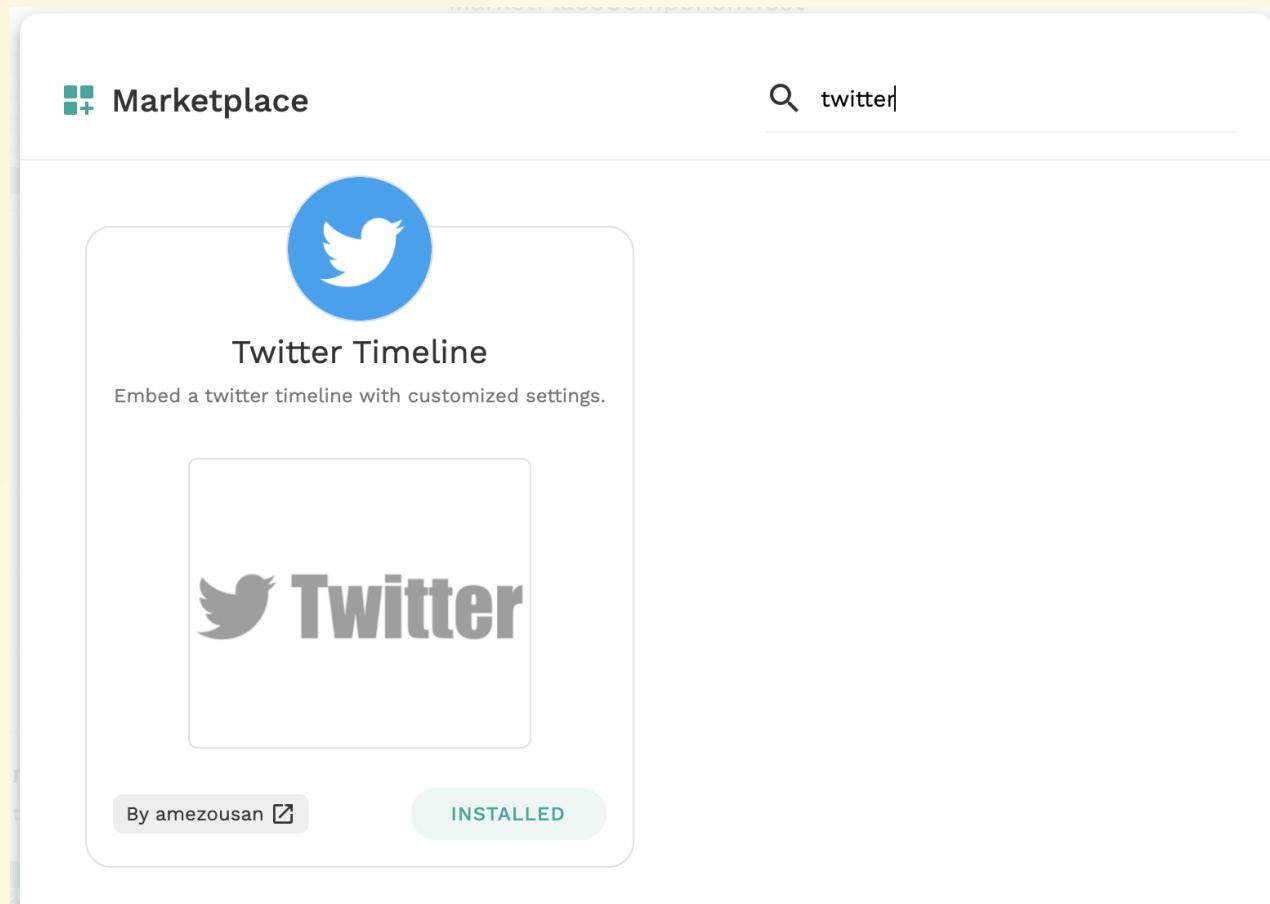
BACK ADVANCED OPTIONS CREATE

# Twitter Timeline Component

- Press the + button and click Explore Marketplace in ADD COMPONENT

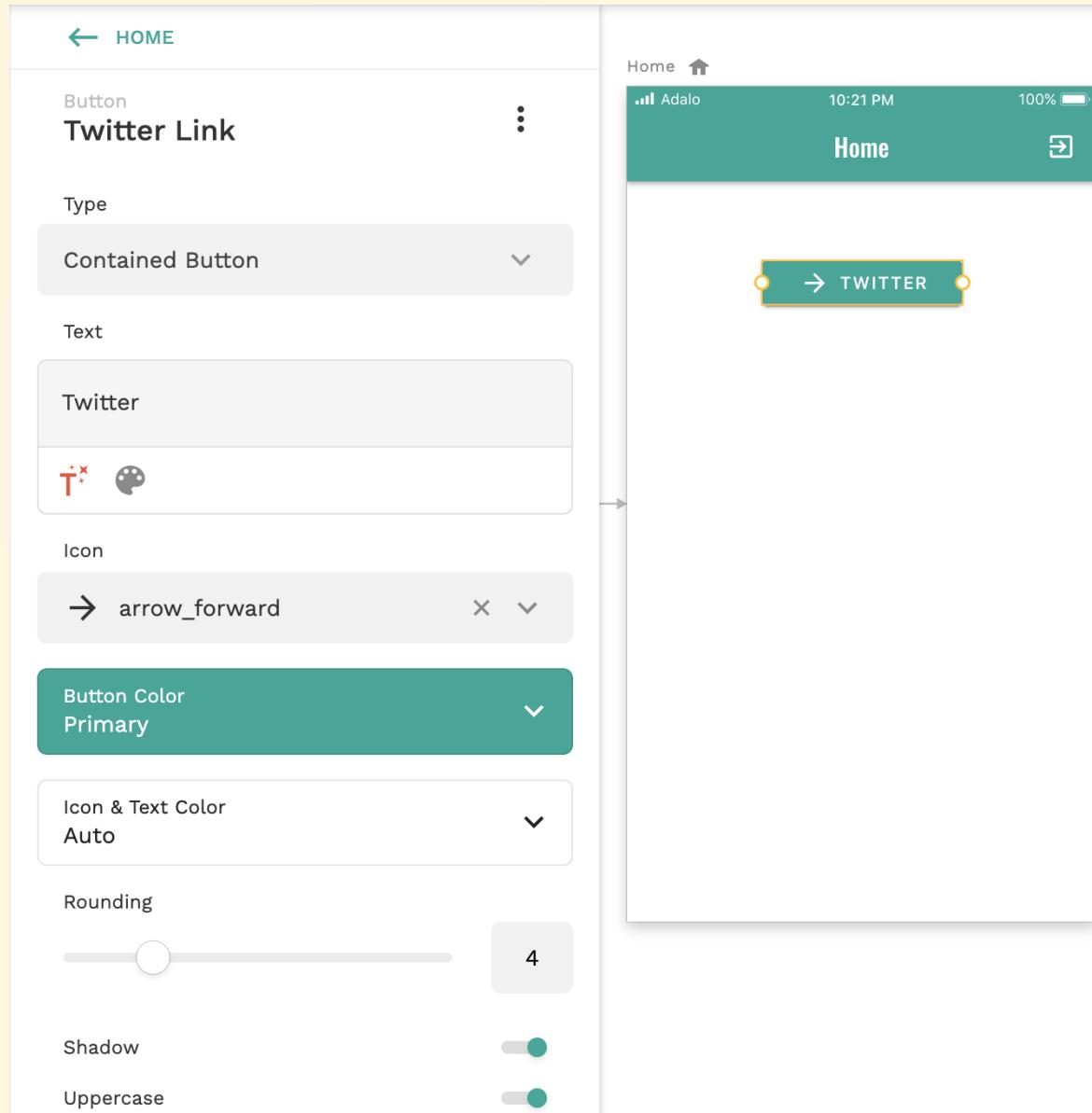


- INSTALL the Twitter Timeline component

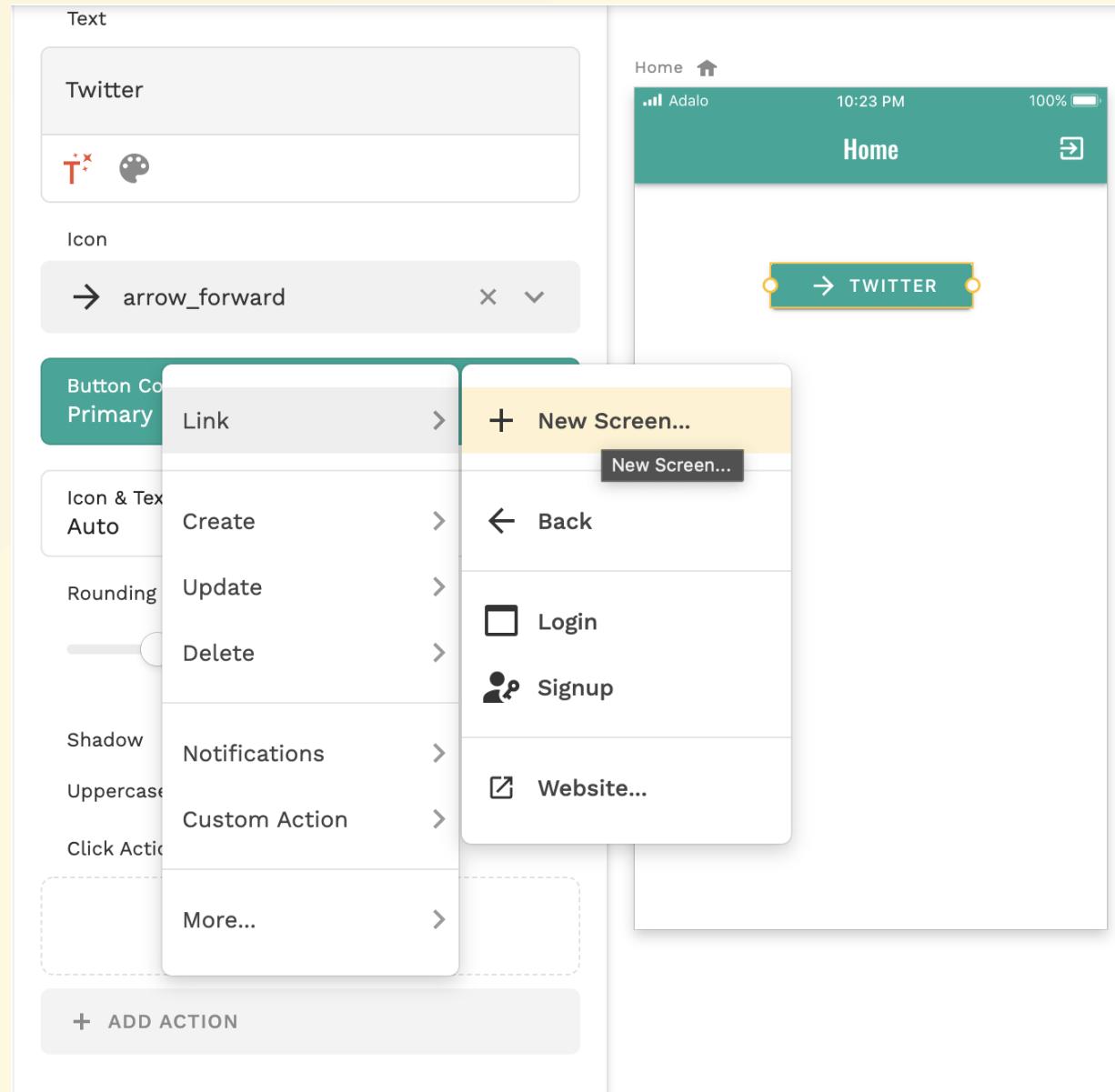


# Let's use Twitter Timeline component.

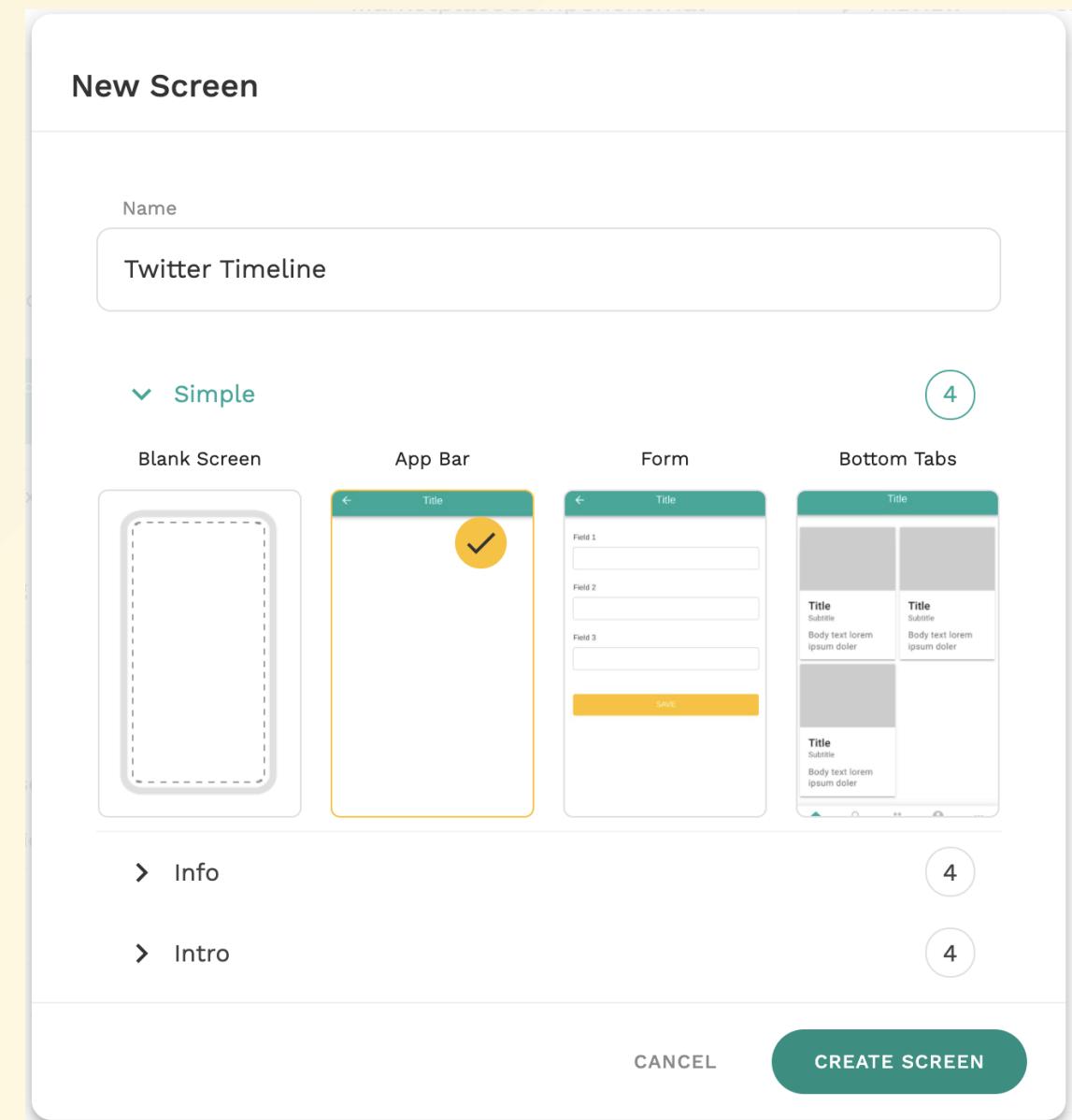
- Add a link button to the Twitter screen to the Home Screen.



- Add Link to New Screen from ADD ACTION



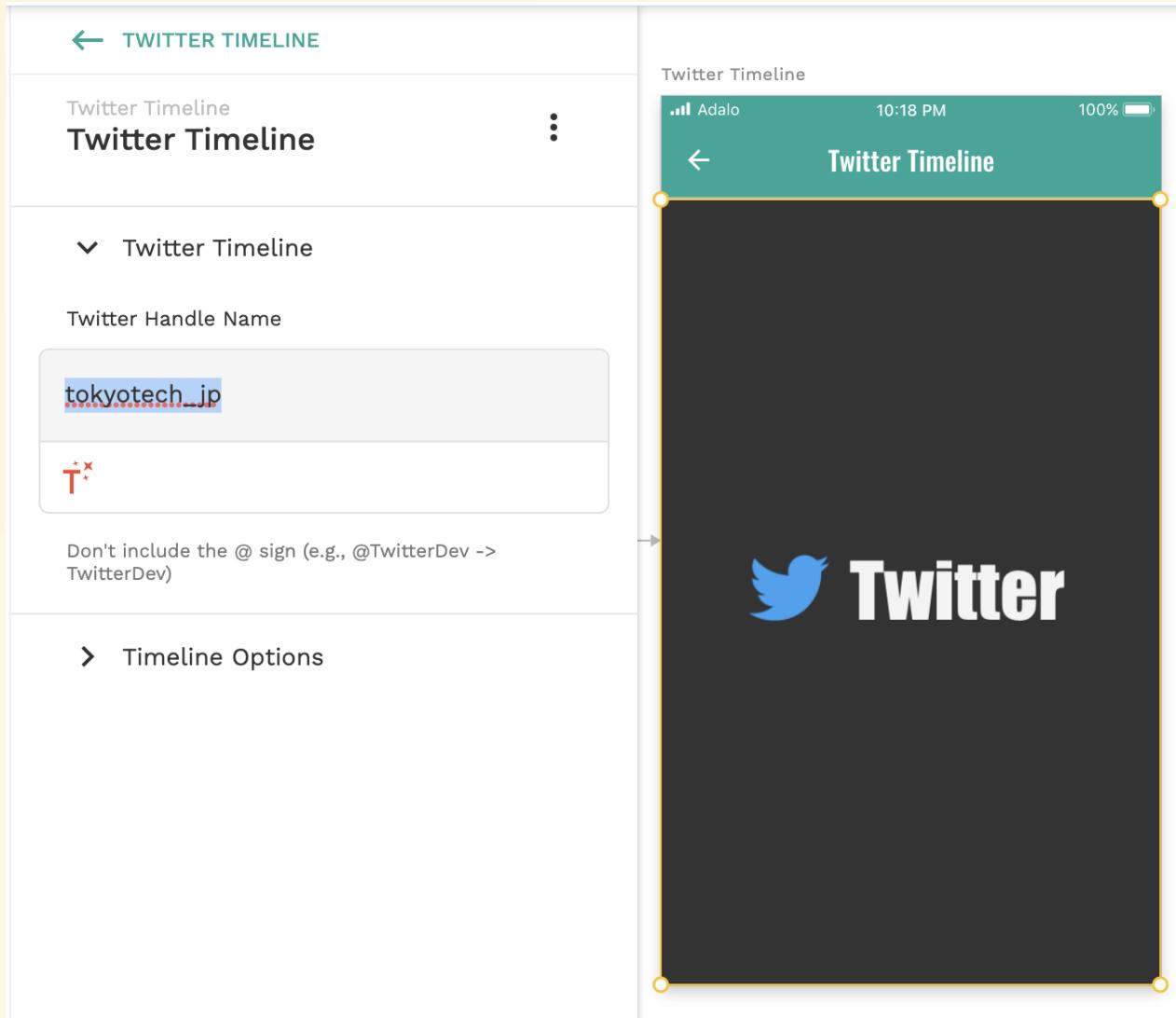
- Select "App Bar" in Template and create a "Twitter Timeline" screen.



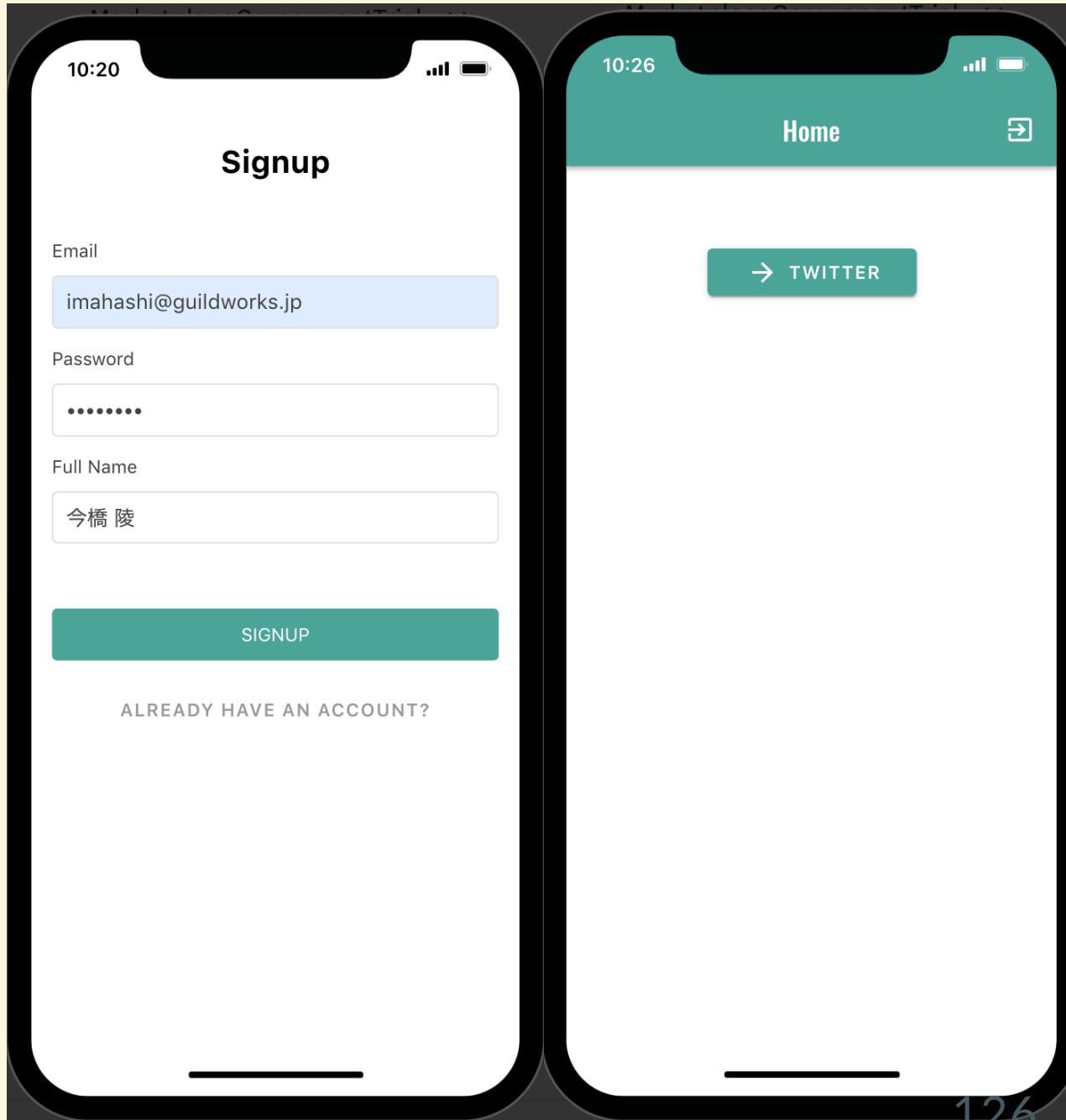
- Place the Twitter Timeline component

The screenshot shows the Adalo component library interface. At the top, there are two navigation items: "Simple" (with a circled '7' icon) and "Forms & Fields" (with a circled '6' icon). Below them is a section titled "Marketplace". A callout box contains the text "More Amazing Components!" and "Check out the Adalo Marketplace to find more components.", with a "EXPLORE MARKETPLACE" button. In the center, there is a section titled "Installed" (with a circled '1' icon) containing a component card for "Twitter Timeline". The card features the Twitter logo and the word "Twitter". To the right of the component card is a preview of the component in a mobile application, showing a teal header with "Twitter Timeline" and a back arrow, and a large black content area below.

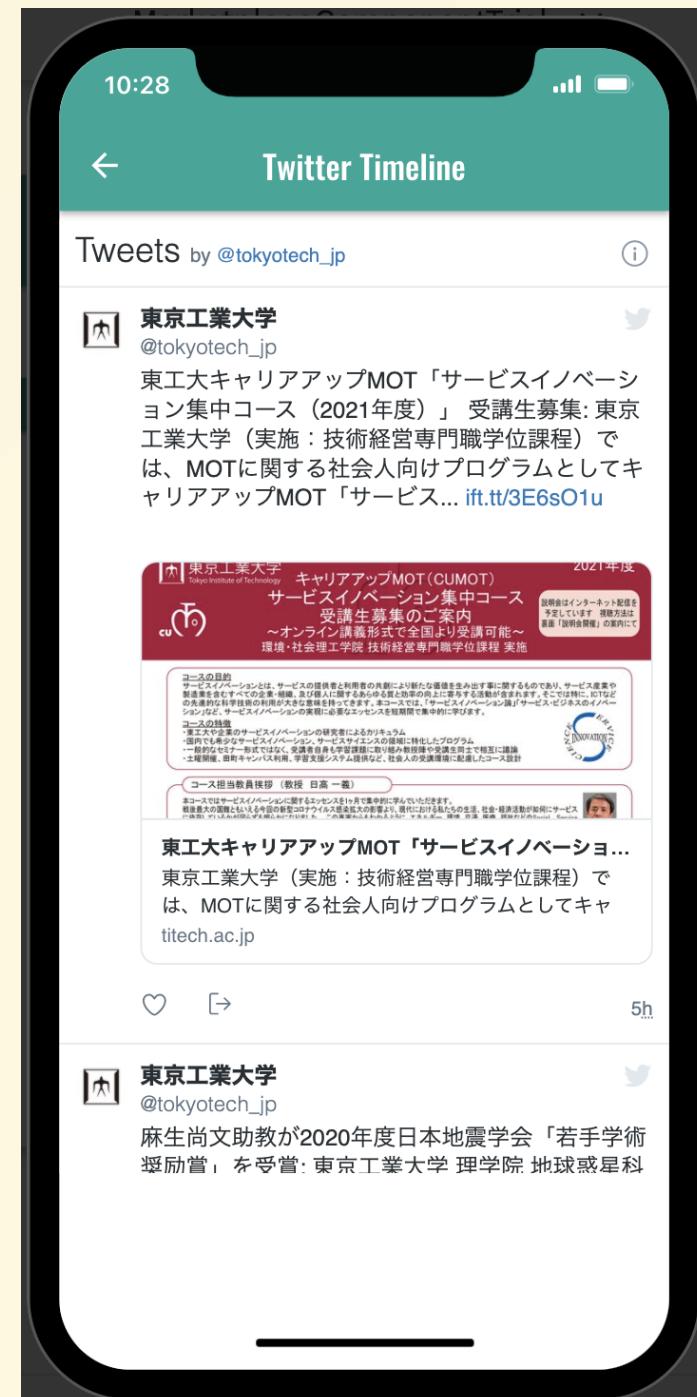
- Enter "tokyotech\_jp" as the Twitter Handle Name
  - You can also change it to the Handle Name of your favorite Twitter account!



- Signup in the Preview function and click the Twitter button.



- The posts of the Twitter account with the Handle Name you entered will be listed.

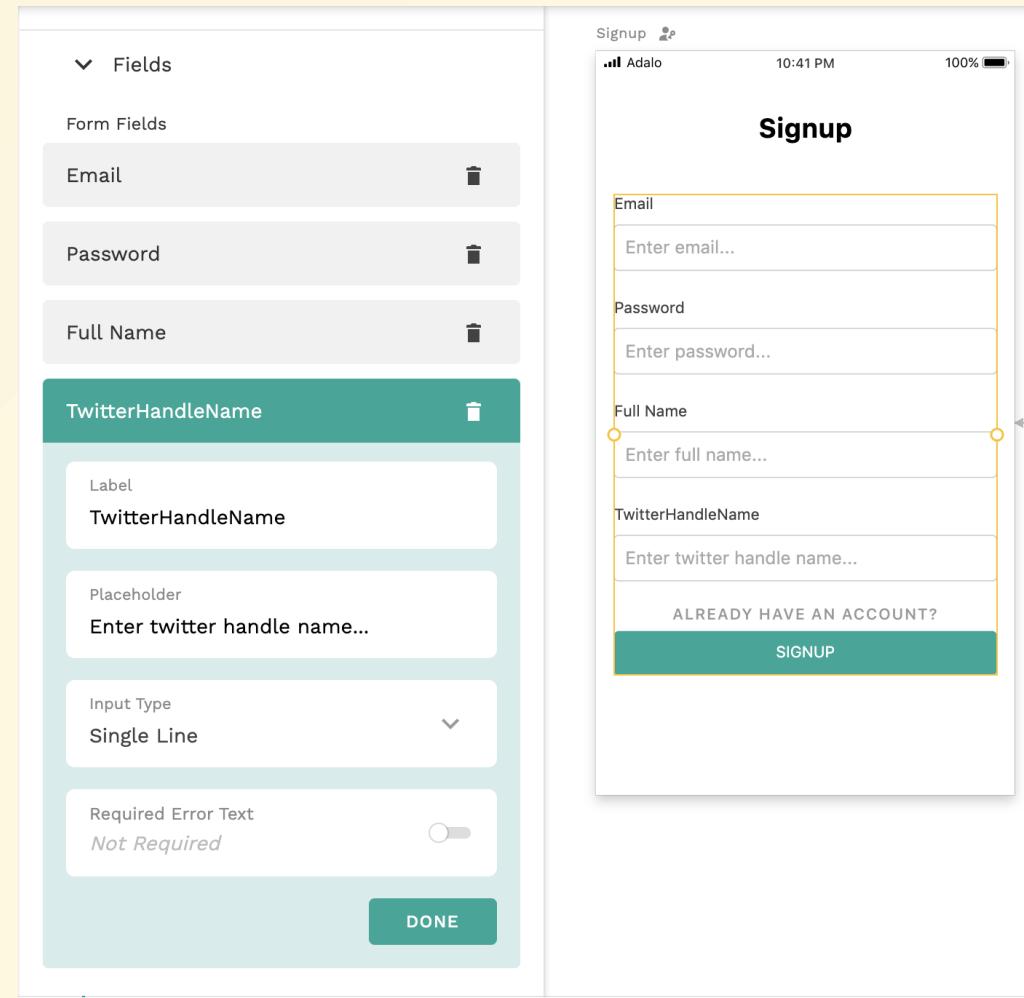


Let's modify it so that it lists the posts of the logged-in user's own Twitter account.

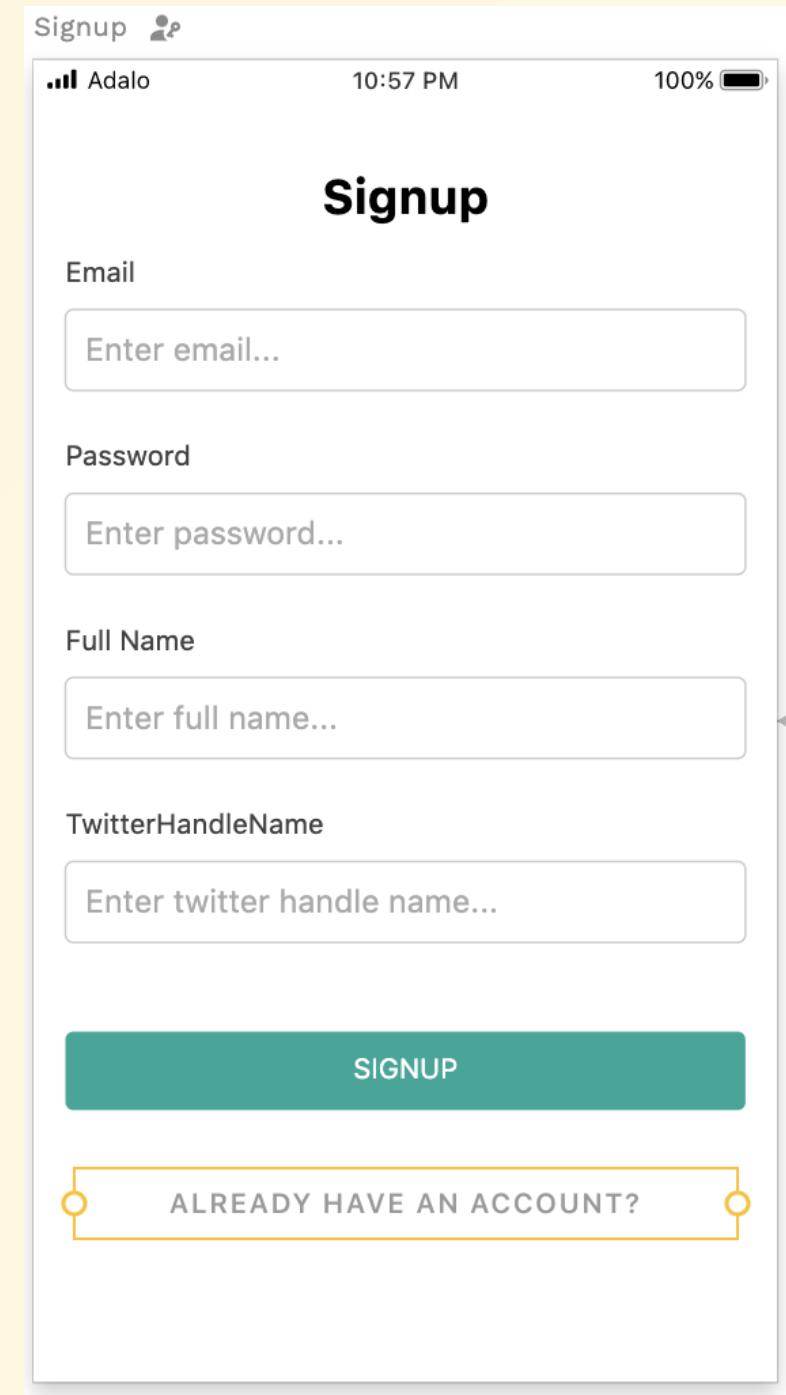
- Add a TwitterHandleName Property to the Users Collection.
  - Select Text as the Type

The screenshot shows a database interface with a sidebar of icons on the left. The main area displays a 'Database Collections' section with a 'Users' collection containing 1 Record. The collection view includes fields for Email, Password, Username, and Full Name. A modal window is open at the bottom, titled 'New Property', showing a 'Type' dropdown set to 'Text' and a 'Name' input field containing 'TwitterHandleName'. At the bottom of the modal are 'CANCEL' and 'SAVE' buttons, and a red 'ADD PROPERTY' button is visible at the bottom right of the main interface.

- Click Form on the Signup screen
- Fields > ADD VISIBLE FIELD > TwitterHandleName
- Change to Not Required

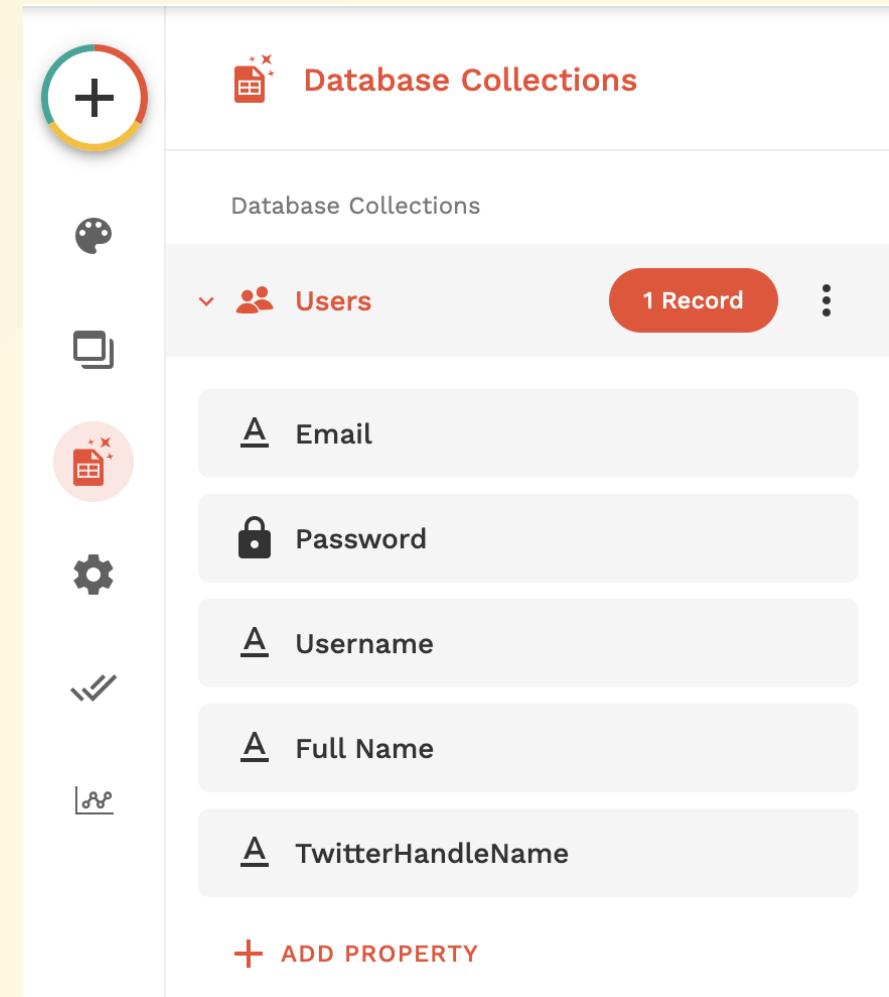


- "ALREADY HAVE AN ACCOUNT?" link is overlapping the form, so move it to the bottom



Let's set the TwitterHandleName for the registered users

- Click the "1 Record" button in the Users Collection



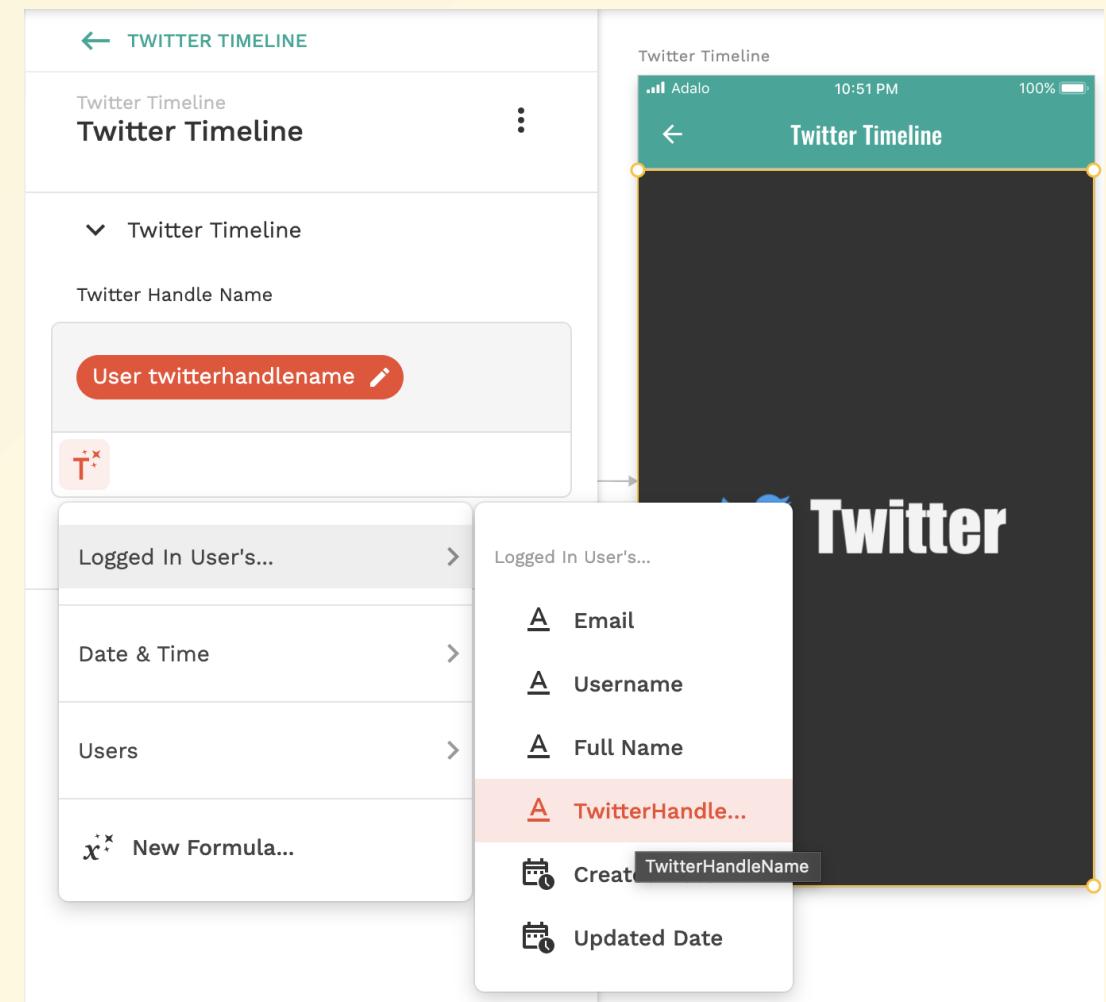
- Click on the record of the registered user.
- Enter TwitterHandleName (You can use your own account or any account you like.)

Jser

A Email	<input type="text" value="imahashi@guildworks.jp"/>
>Password	<input type="password"/>
A Username	<input type="text"/>
A Full Name	<input type="text" value="今橋 陵"/>
A TwitterHandleName	<input type="text" value="titechcoop"/> <span style="color: red;">.....</span>

CANCEL S

- Change the Twitter Handle Name in the Twitter Timeline component to "Logged In User's TwitterHandleName".

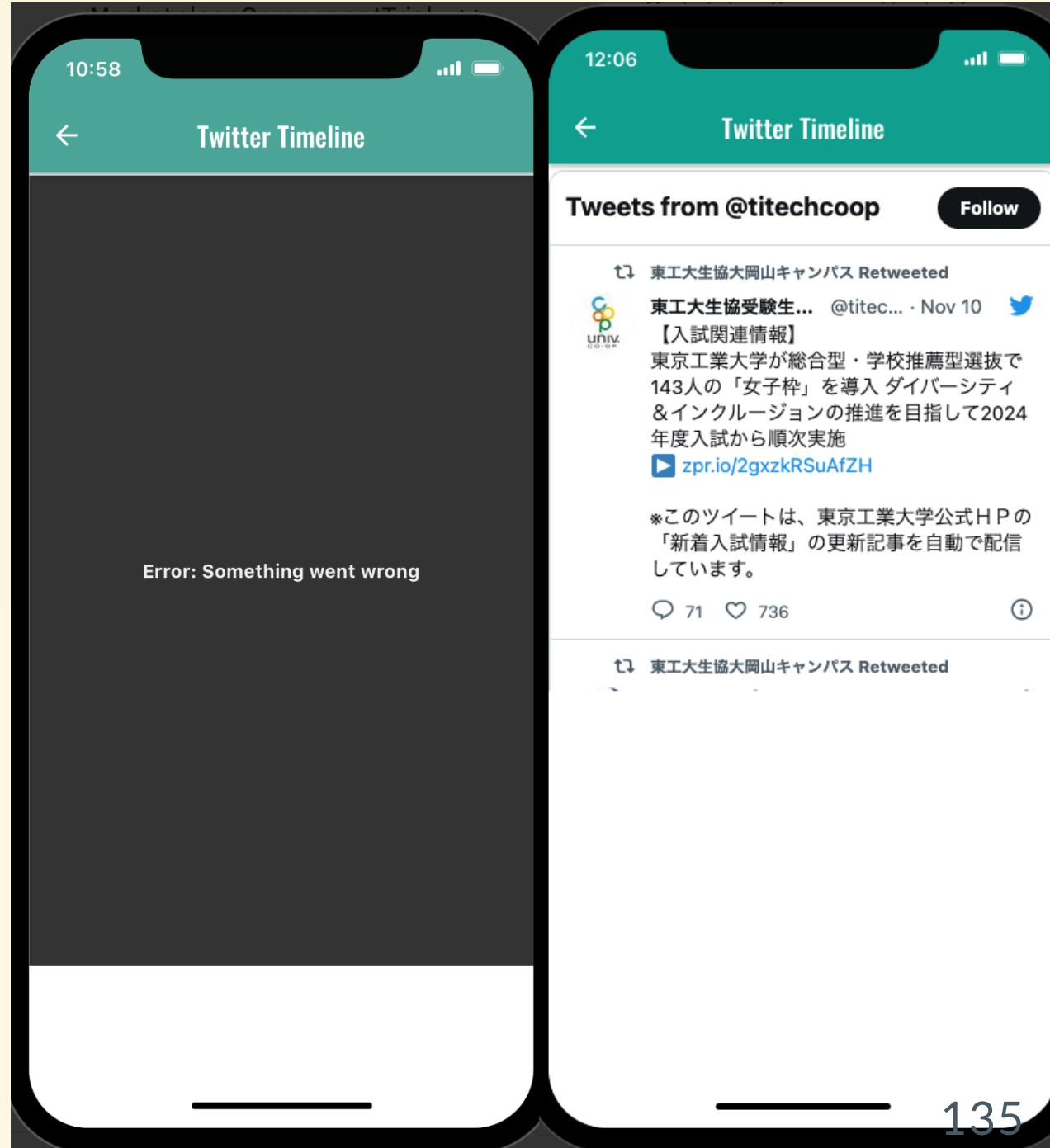


Theoretically, this should display the list of posts of the Twitter account set by the logged-in user himself.

Let's check it out with the Preview function.

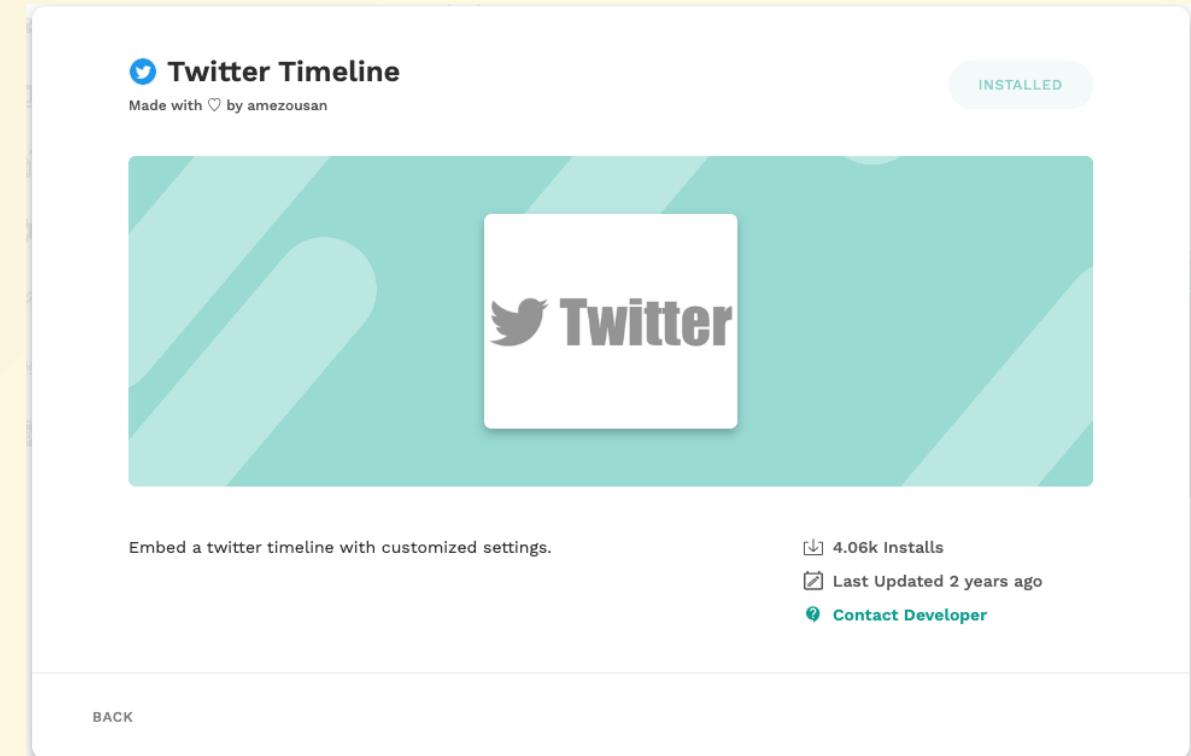
I think an error will occur at first.

If so, go home screen by clicking "←" icon. After that, open "Twitter Timeline" screen again. You can see the posts of the logged-in user's own Twitter account.



Twitter Timeline is a component that has been made available for free by volunteers.

What someone else makes doesn't always work the way you want it to.



There are several other components provided to integrate with external services. If you are interested, you can try them.

Example:

- Youtube (free)
- Google Map(free trial is available by registering credit card)

## Custom Action

Next to the external integration component, I will show you how to handle the data retrieved from the API on the Adalo screen.

## Reference

“ In a broad sense, an Application Programming Interface (API) is a specification of an interface used by software components to exchange information with each other. ”

<https://ja.wikipedia.org/wiki/アプリケーションプログラミングインターフェース>

First, let's try API integration.  
We will use The Cat API, which is free to try. Please access the following URL.

<https://thecatapi.com/>

The screenshot shows the homepage of The Cat API. At the top, there is a navigation bar with links for PRICING, ABOUT, and DOCUMENTATION. Below the navigation bar, the main heading reads "The Cat API - Cats as a Service." A sub-headline says "Because everyday is a Caturday." Below this, a text block states: "A public service API all about Cats, free to use when making your fancy new App, Website or Service. [Get your license now.](#)" To the right of the text, there is a video player displaying a cat walking on a wooden floor. Above the video player are two buttons: a green "LOVE IT" button with a thumbs-up icon and a red "NOPE IT" button with a thumbs-down icon. At the bottom left, there is a footer banner with the text "The Cat API has delivered Billions of requests, to over 30K developers". On the left side of the page, there are icons for various technologies: React, jQuery, Vue.js, Node.js, and Angular.

Note: For the dog people, there is also The Dog API. It probably does the same thing as The Cat API. (I have not tried it, so I recommend using The Cat API with it first)

<https://www.thedogapi.com/>

The screenshot shows the homepage of The Dog API. At the top, there's a blue header bar with navigation links: PRICING, ABOUT, and DOCUMENTATION. Below the header, the main title "The Dog API - Dogs as a Service" is displayed, followed by the tagline "Because everyday is a Dog Day." A short description states: "A public service API all about Dogs, free to use when making your fancy new App, Website or Service. [Get your license now.](#)" Below the text, there are five developer logo icons: React, jQuery, Vue.js, Node.js, and Angular. The central feature is a large image of a white dog sitting in a green field. Above the image are two buttons: a green "LOVE IT" button with a thumbs-up icon and a red "NOPE IT" button with a thumbs-down icon. At the bottom of the page, a blue footer bar contains the text: "The Dog API has delivered Billions of requests, to over 30K developers".

When using an API, it is often necessary to have an API key issued by the API provider. The Cat API also requires an API key, so have it issued to you.

- Scroll down to the bottom and click "SIGNUP FOR FREE" in the Pricing section.

## Pricing

Free



\$0

No hidden costs

Free for you to use on your non-monetized app

SIGNUP FOR FREE

- 10,000 requests / month
- Free Code Samples
- [Community forums](#)
- SSL Encryption

- Enter your email, App Description, type of project and click "SIGNUP" !

## Signup for an API key

It's completely free, email is only used to send you an API Key & your stats, and you can use the API as much as you like. No Spam Ever.

E-mail

 your-email-address@example.com

App Description

 Sample App

What type of project will you use the API for?

A personal project

A project for school/college/university

A business project

Opt-in to my newsletter about other APIs I'm building?

**SIGNUP**

[Privacy Policy](#) | [Terms & Conditions](#)

- Check the API key that was sent to you by email (you will use it later).

The screenshot shows an email client interface with a single message. The message is from 'aden@thatapiguy.com' to 'aden@thatapiguy.com' (To: 自分). The subject of the email is 'The Cat API - API Key'. The email body contains the following text:

Welcome to The Cat API!

Thanks for signing up, welcome to the API!

Your API key:

xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx

Use it as the 'x-api-key' header when making any request to the API, or by adding as a query string parameter e.g. 'api\_key=xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx' [More details on authentication.](#)

[API Documentation](#) | [Postman Collection](#) | [Public Forum](#)

Join our new API Slack Group - [That API Slack Group](#)

All the best, Aden & That API Team.

p.s. We build APIs for companies & good causes. If you're interested find out more at <https://thatapicompany.com>

Next, let's check the API documentation to see how to use the API.

- Go to the following URL (you can also find the link "API Documentation" in the email you received earlier)

<https://docs.thecatapi.com/>

Use the API to get a random kitten image as described in the top page.

As in the example, make sure that the image changes when the button is pressed.

#### Quickstart - Get a random Kitty

Using the language / framework of your choice:

1. Load <https://api.thecatapi.com/v1/images/search>
2. Get the first Array object of the JSON response
3. Load it's .url
4. Enjoy the kitty 😺

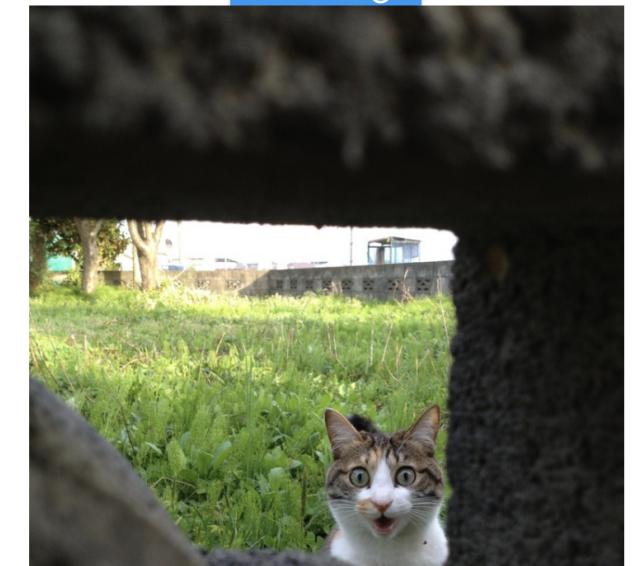
Bonus: Add a button to repeat the request for a new kitty!

#### Example

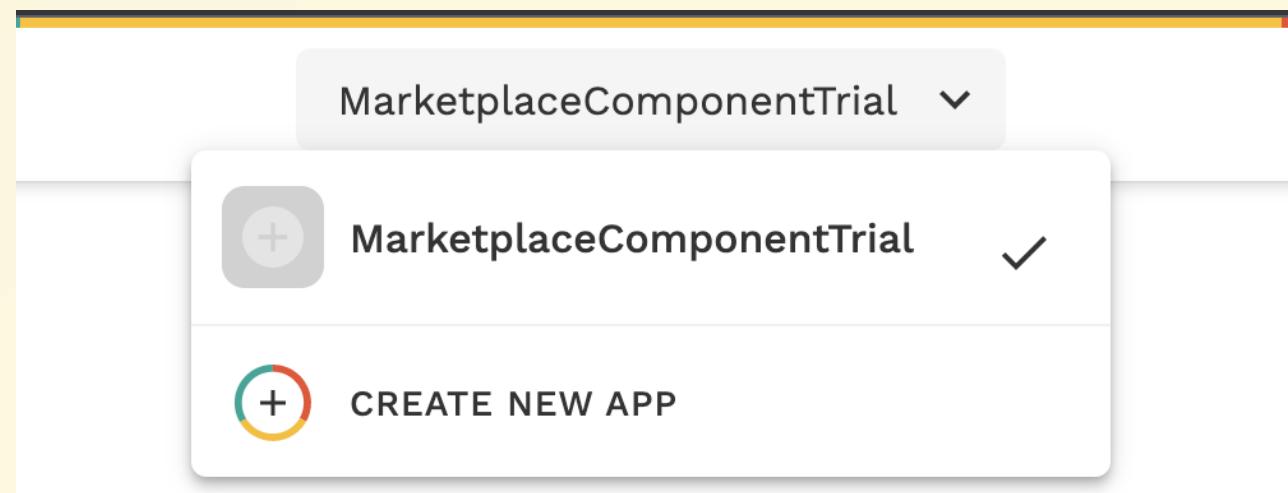
Uses <https://api.thecatapi.com/v1/images/search> to show one random image every time "Another" is clicked.

Example Source Code: VueJS

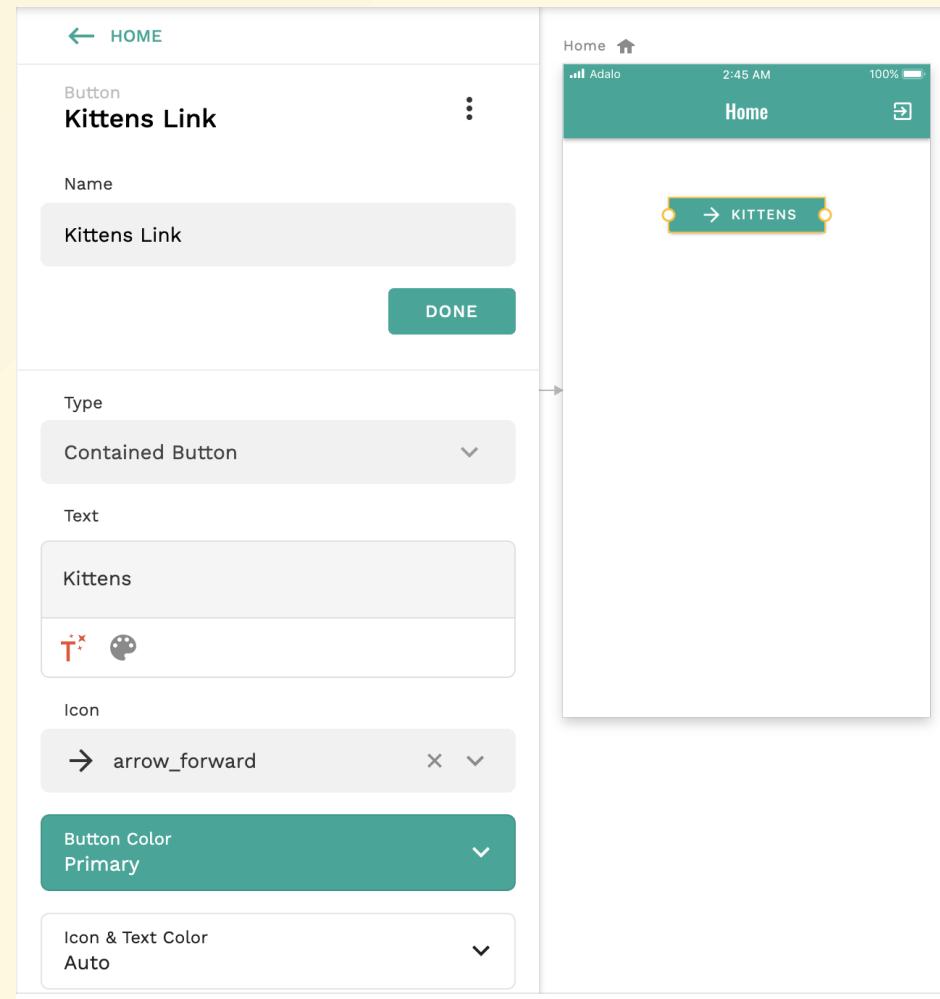
ANOTHER 



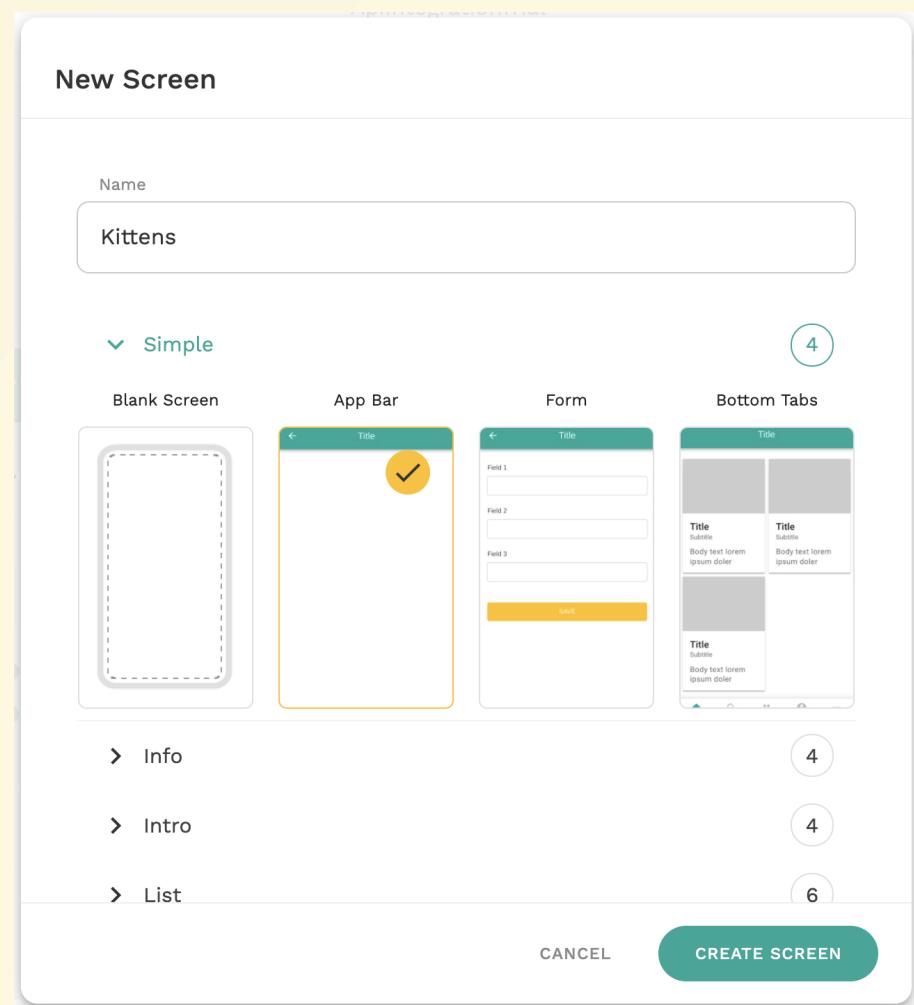
- CREATE NEW APP in Adalo's admin panel
- The settings are as follows
  - Platform: Native Mobile App
  - Template: Blank
  - App Name: ApilntegrationTrial



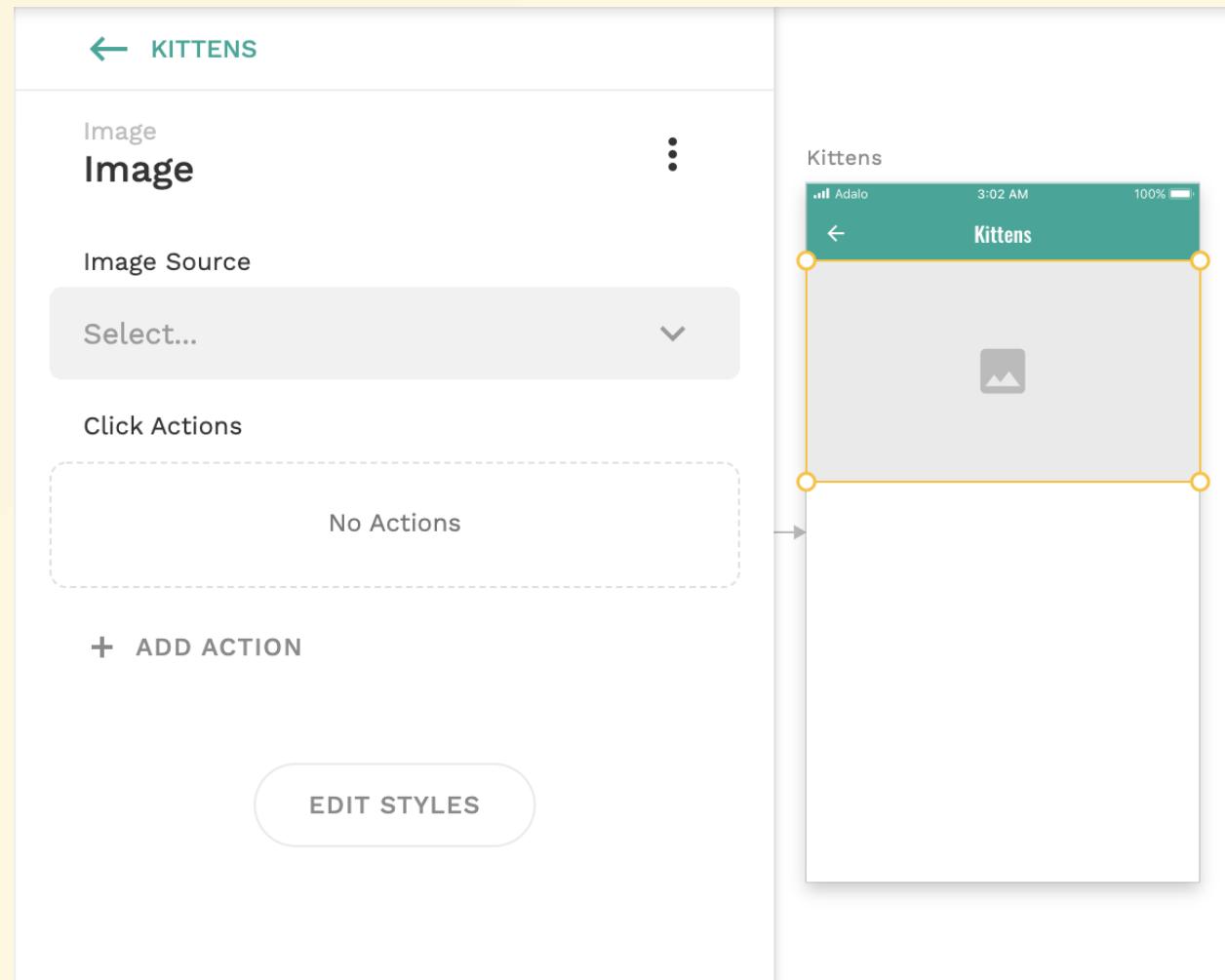
- Add a link button to the kitten image display screen on the Home screen.
- Set Link to New Screen from ADD ACTION



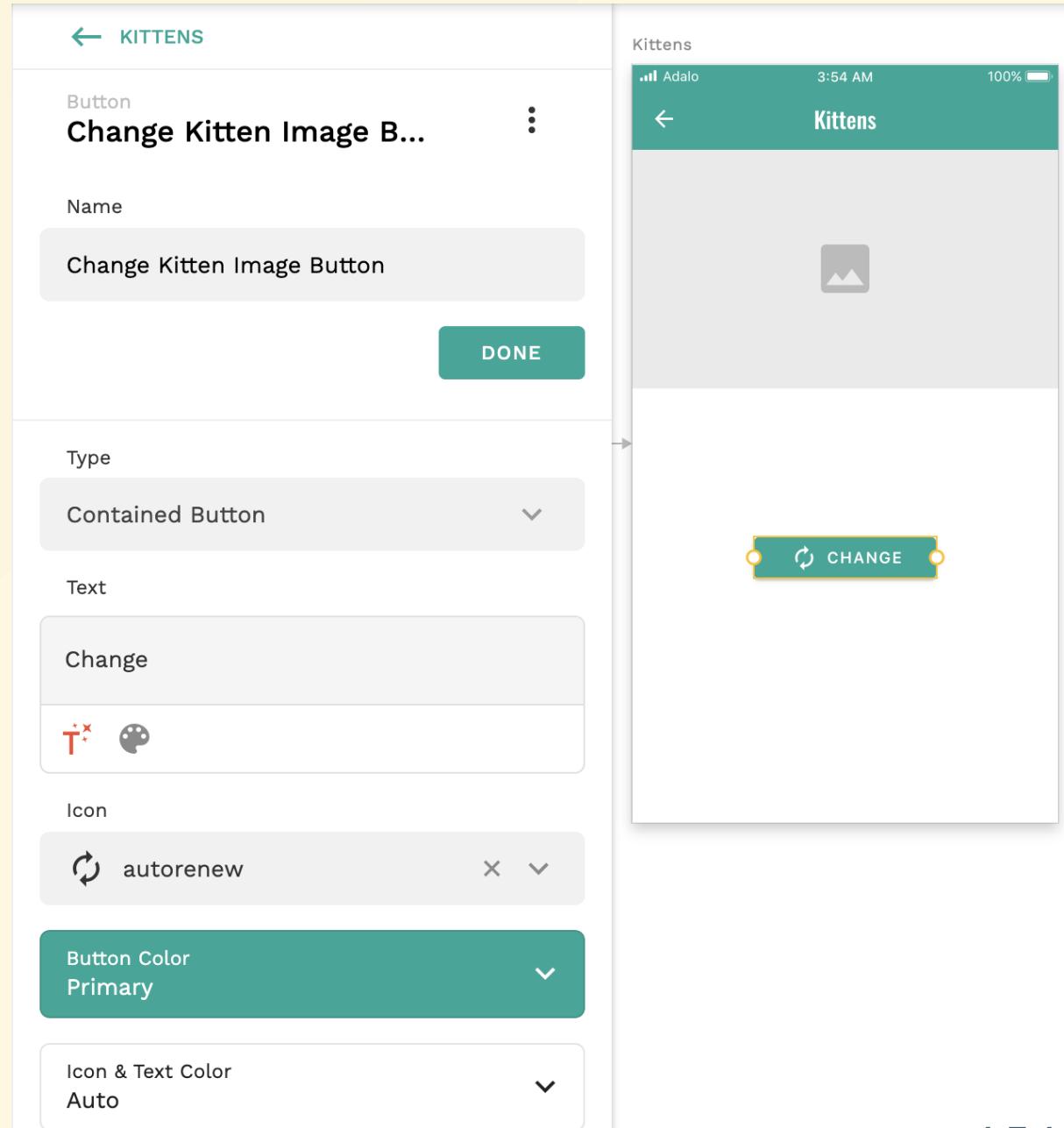
- Select App Bar as Template and create Kittens screen.



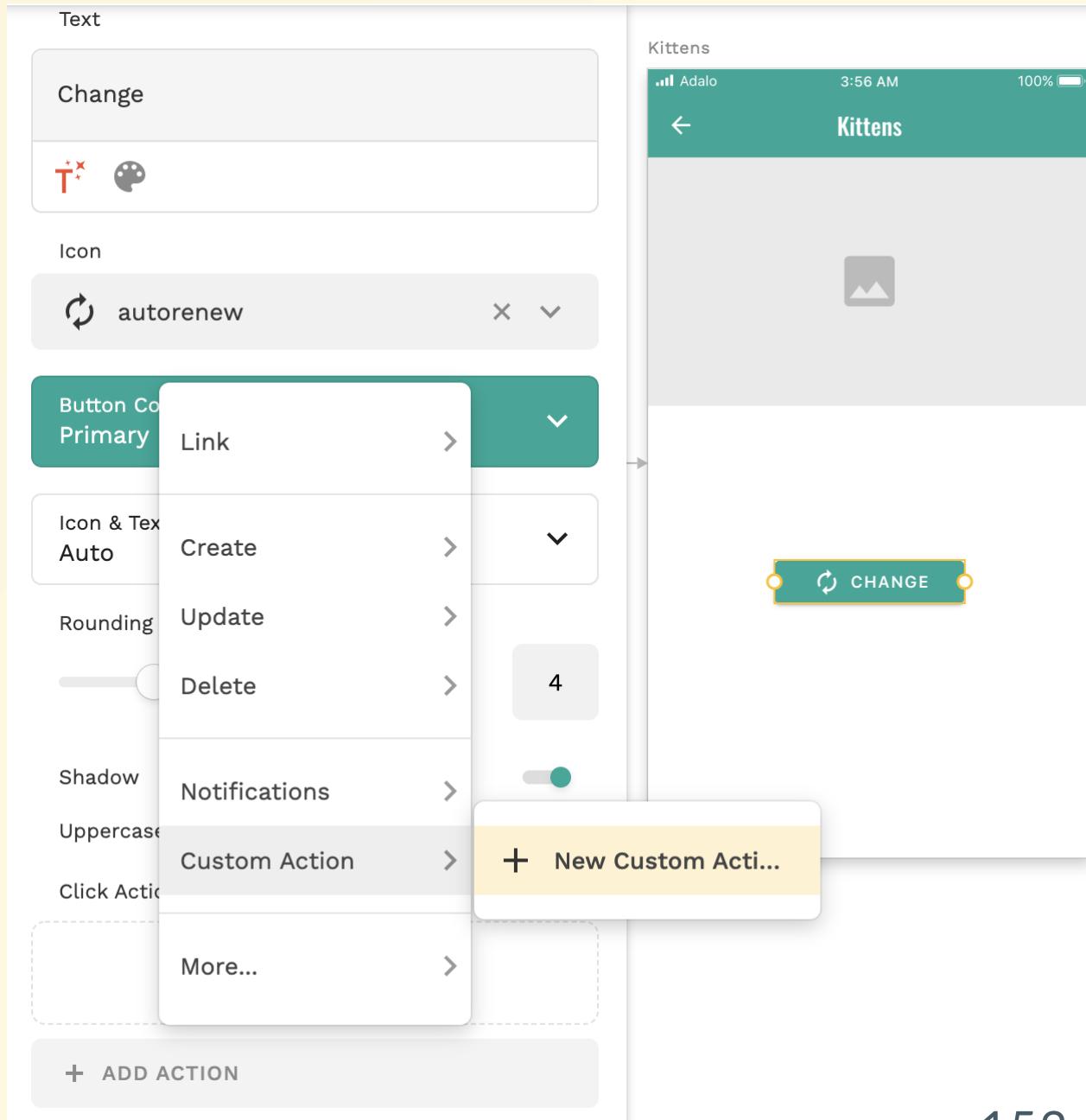
- Place the Image component on the screen
- Leave the component settings as they are (we'll set them later)



- Add a Change Kitten Image Button

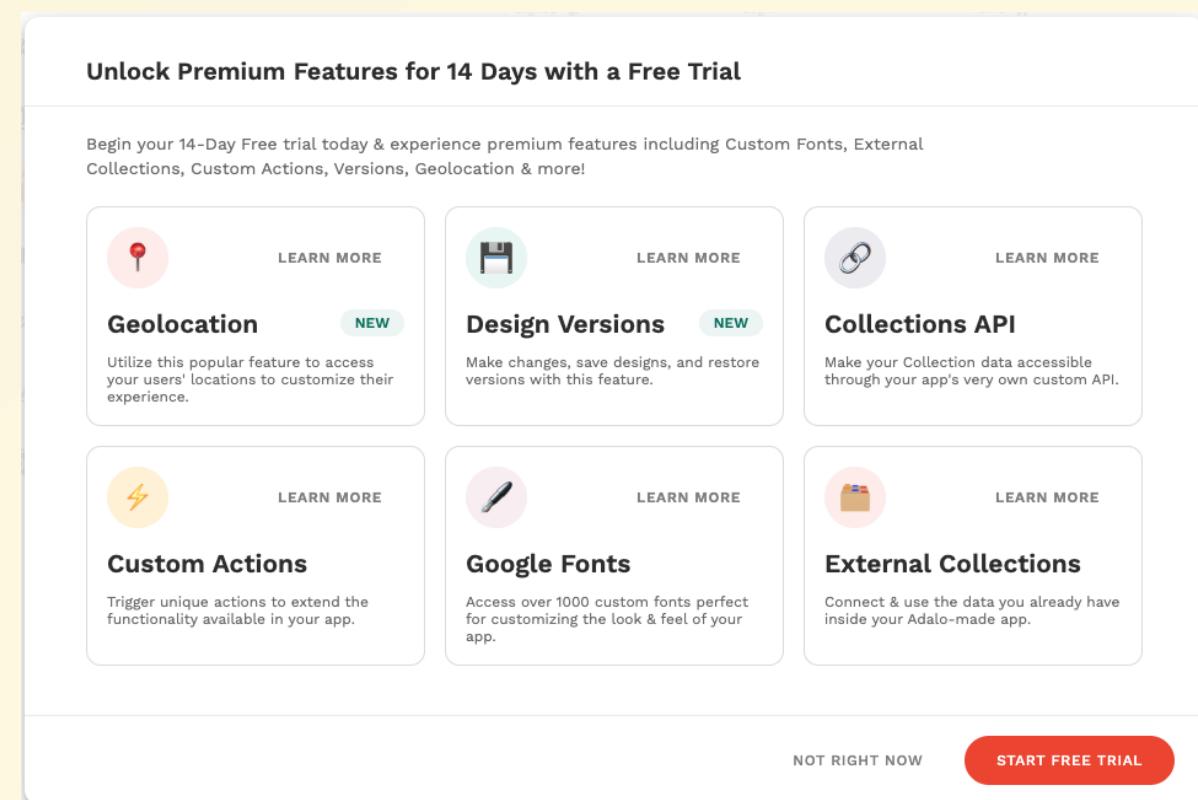


- Select New Custom Action from ADD ACTION



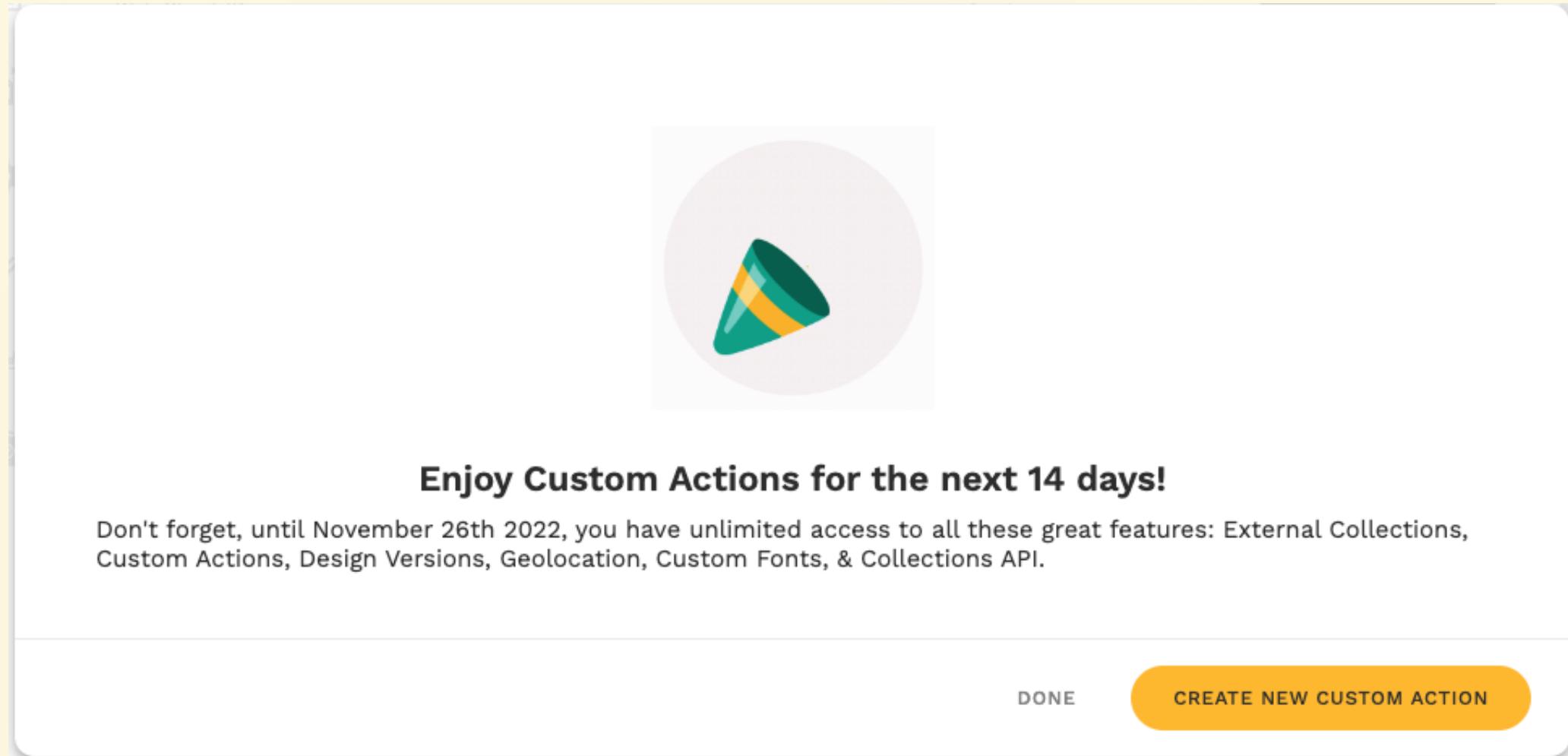
You will be prompted to start a 14-day Free Trial.

- Click on "START FREE TRIAL"



The trial has started.

- Click on "CREATE NEW CUSTOM ACTION" !



- Enter the following and click NEXT
  - Name: GetRandomKitten
  - Type: Create

The screenshot shows a user interface for creating a new item. It consists of three horizontal steps: 1. Name & Type (highlighted in yellow), 2. API Request, and 3. Outputs. Step 1 is completed, showing the name "GetRandomKitten" and type "Create". Step 2 is partially visible below. At the bottom right are "CANCEL" and "NEXT" buttons.

1 Name & Type ————— 2 API Request ————— 3 Outputs

Name  
GetRandomKitten

Type  
Create

CANCEL      NEXT

# Next, we will configure the API Request to be sent.

The screenshot shows a configuration interface for an API request. At the top, there are three numbered steps: 1. Name & Type (done), 2. API Request (current), and 3. Outputs. The API Request step contains the following fields:

- API Base URL:** https://api.example.com/v1/action
- Method:** Select a Method (dropdown menu)
- Headers & Queries:** No Auth Parameters
- Body:** Enter JSON here

On the right side, under the heading "Inputs", it says "No Inputs" and has a "+ ADD ITEM" button.

At the bottom right of the main area is a yellow button labeled "RUN TEST REQUEST". Below the main area are "BACK" and "NEXT" buttons.

Go to the following URL to check the configuration items.

<https://docs.thecatapi.com/api-reference/images/images-search>

- API Base URL is  
<https://api.thecatapi.com/v1/images/search>
- Method is GET
- Set the API key in the Header with the name x-api-key

## GET /images/search

Search & Iterate through all public images. <https://api.thecatapi.com/v1/images/search>

### Authorization

#### apiKey - x-api-key

Just use the API Key emailed to you when you signed up.

- Name: x-api-key
- Location: header

- Set the API Request based on the results of the check.
  - API Base URL is <https://api.thecatapi.com/v1/images/search>
  - Method is GET
  - Set the API key in the Header with the name x-api-key
- Click "RUN TEST REQUEST".

The screenshot shows a user interface for configuring an API request. At the top, there are three tabs: 'Name & Type' (yellow), '2 API Request' (orange), and '3 Outputs' (grey). The 'API Request' tab is active.

**API Base URL:** https://api.thecatapi.com/v1/images/search

**Method:** GET

**Headers & Queries:**

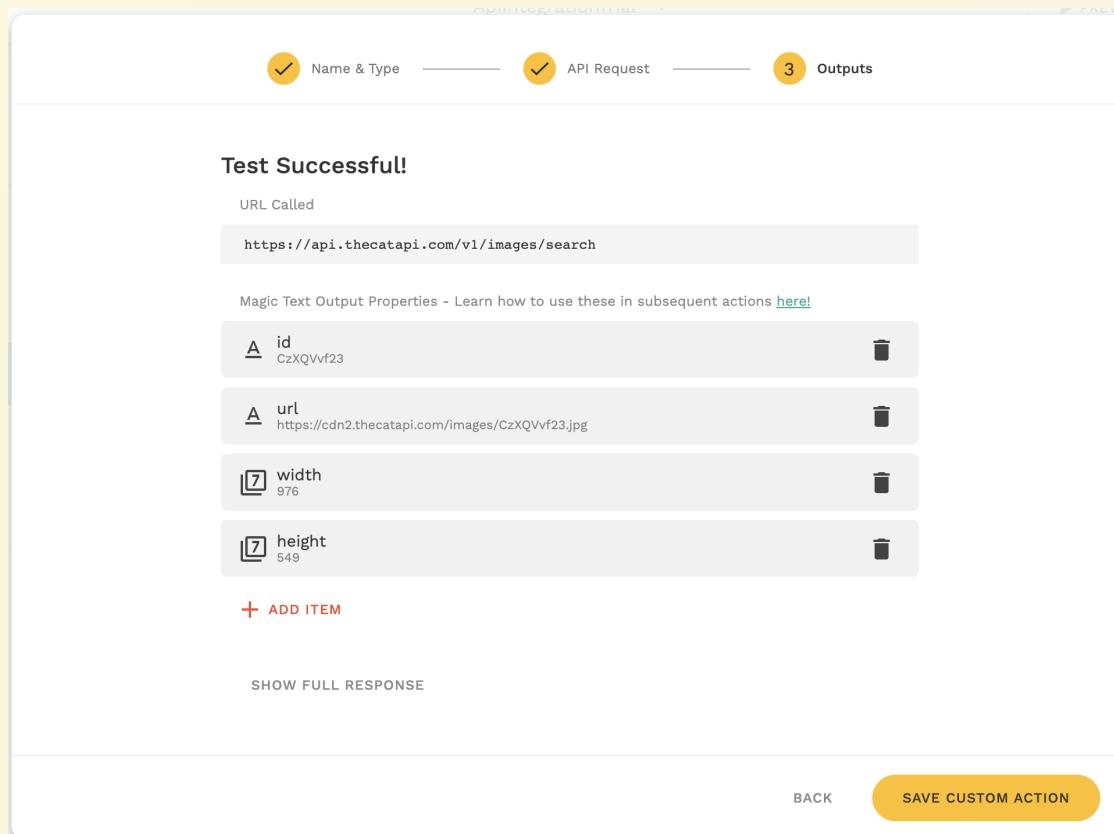
- Header:** x-api-key
- Name:** x-api-key
- Value:** 250736df-9bd0-40c6-9090-5f16b5b8ef05

**Inputs:** No Inputs

**Buttons:** BACK, RUN TEST REQUEST, DONE

If the Test is successful, the data retrieved from the API (Magic Text Output Properties) will be displayed. These can be used in subsequent actions.

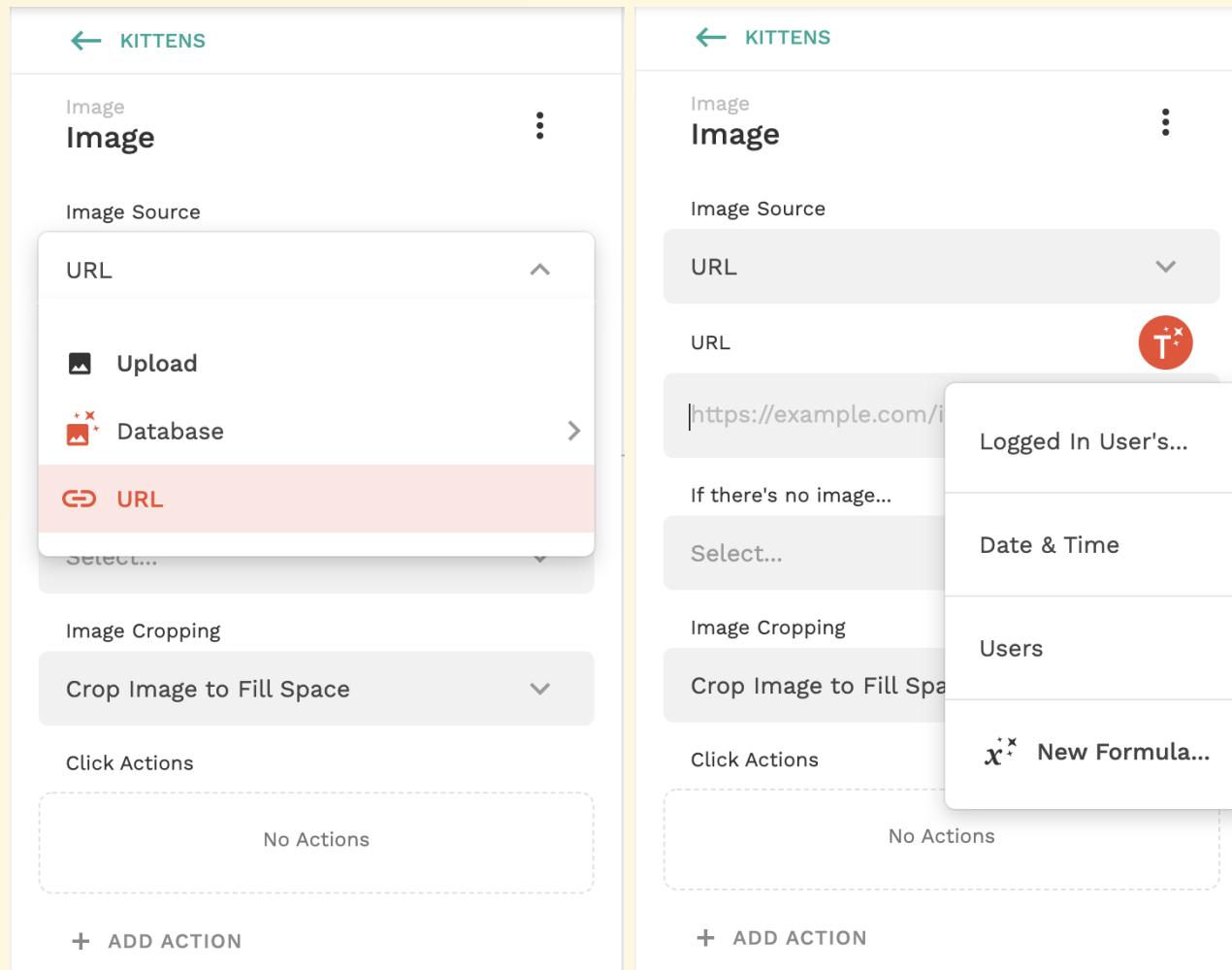
- Click on "SAVE CUSTOM ACTION"



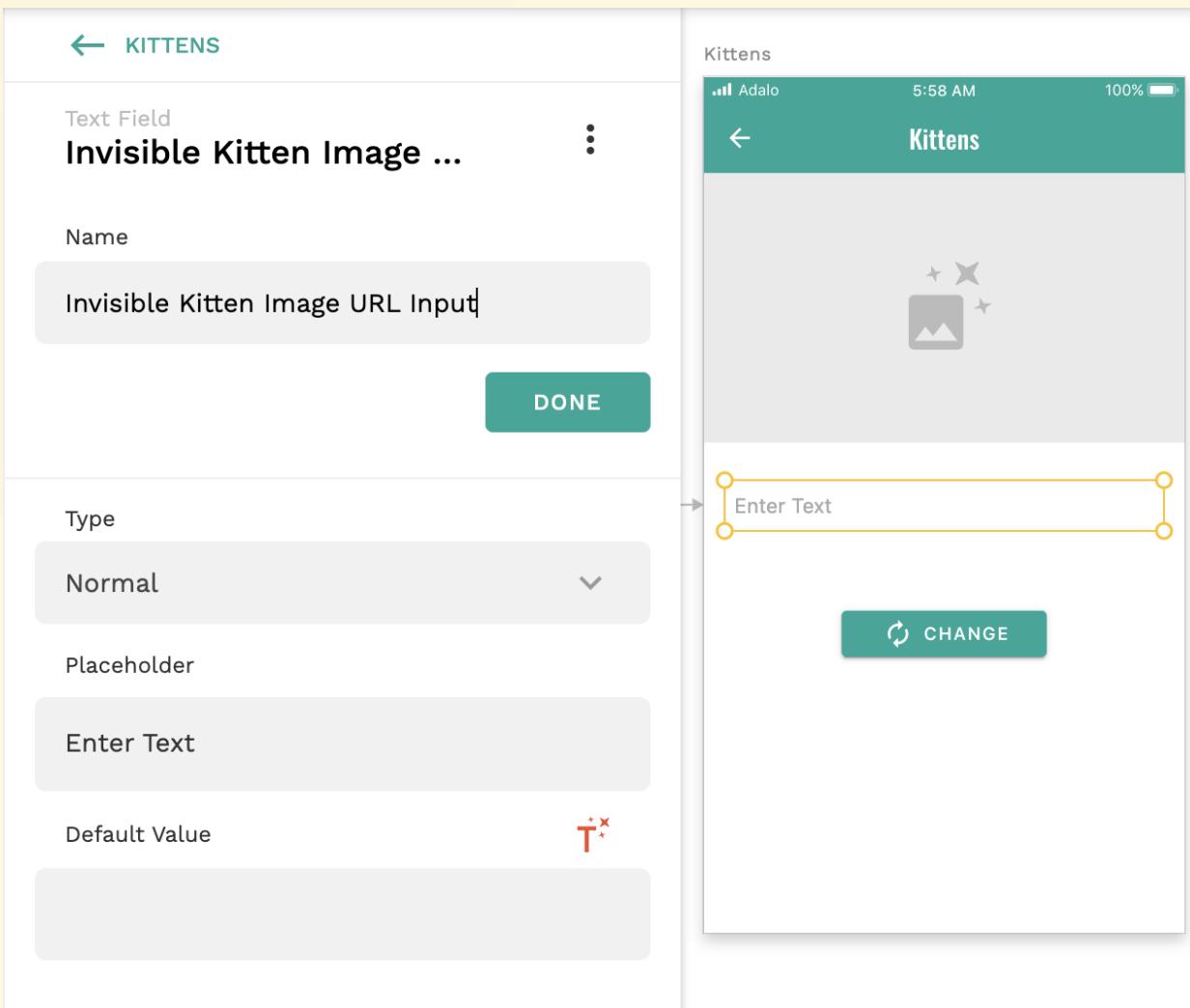
Next, set the URL of the kitten image retrieved from the API to the Image Source of the Image component.

As it is, the data retrieved from the API will not appear in the choices.

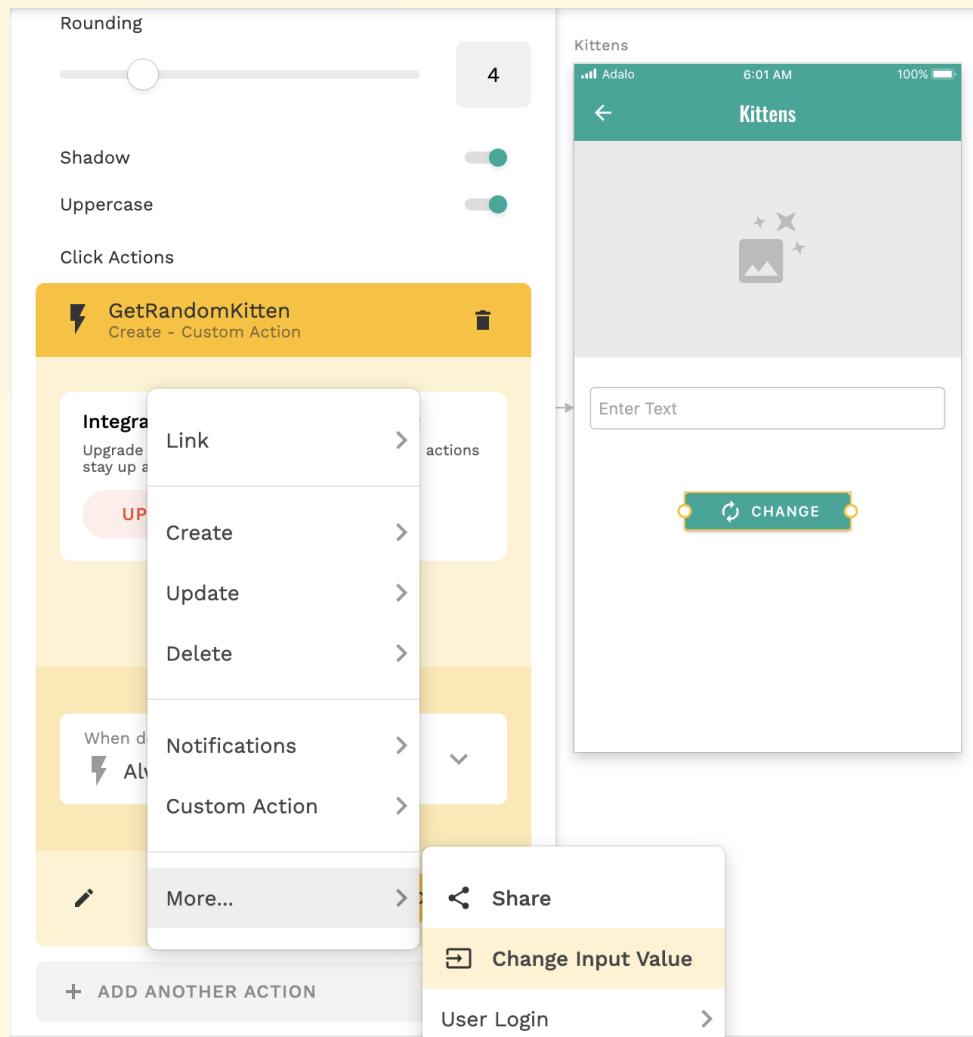
A url  
<https://cdn2.thecatapi.com/images/6k7.jpg>



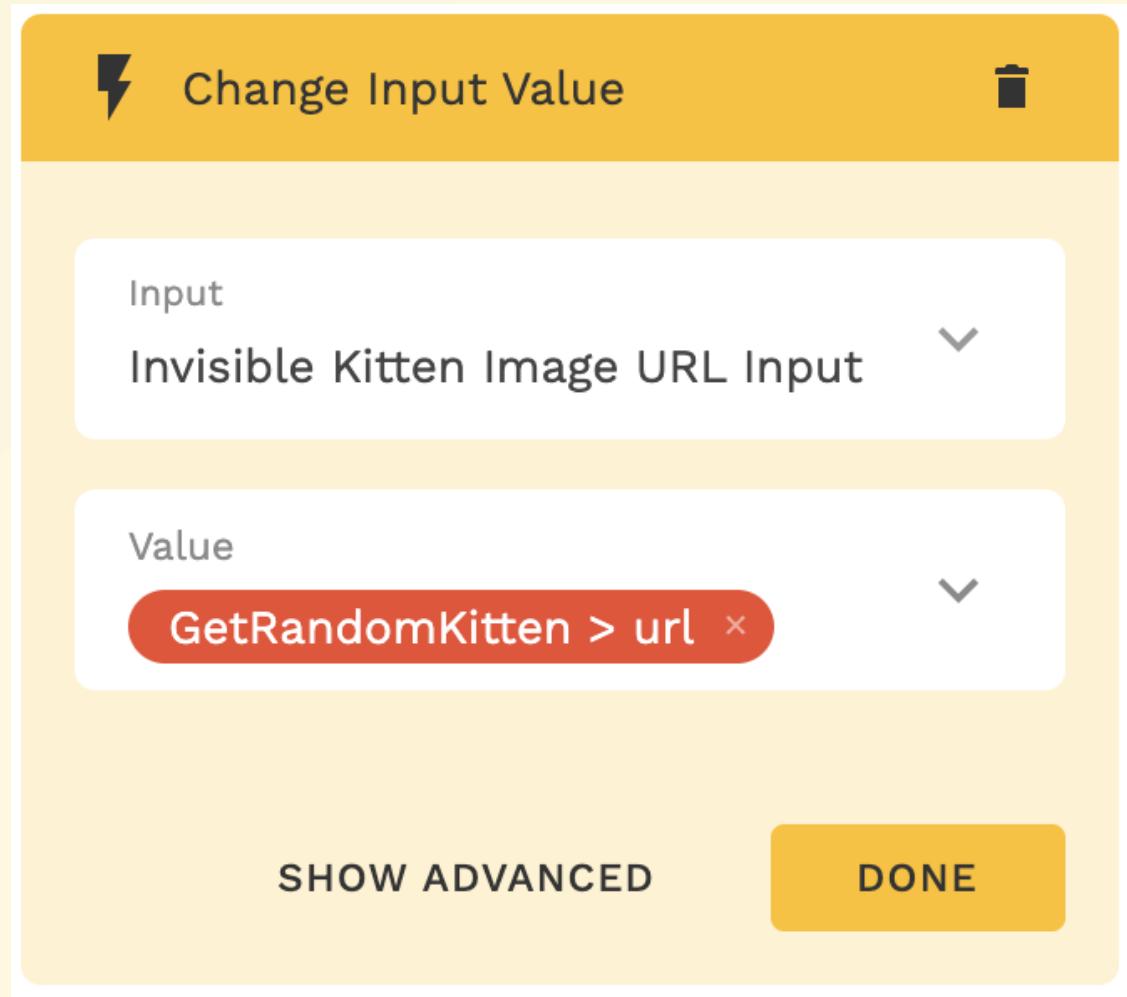
- Add a Text Input component to the screen
- Change the Name to "Invisible Kitten Image URL Input".



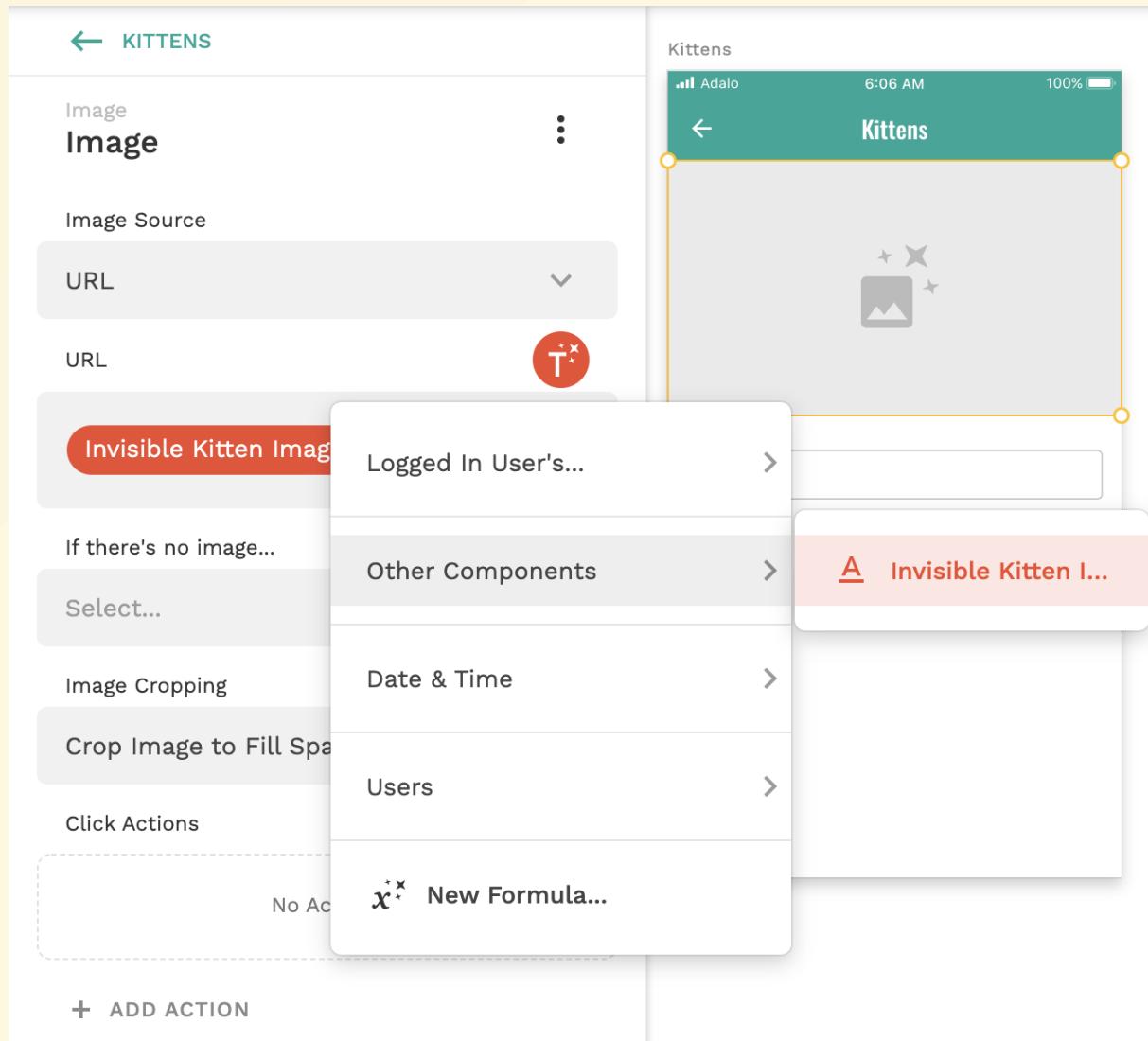
- Click on the "Change Kitten Image Button"
- Select "Change Input Value" from ADD ANOTHER ACTION



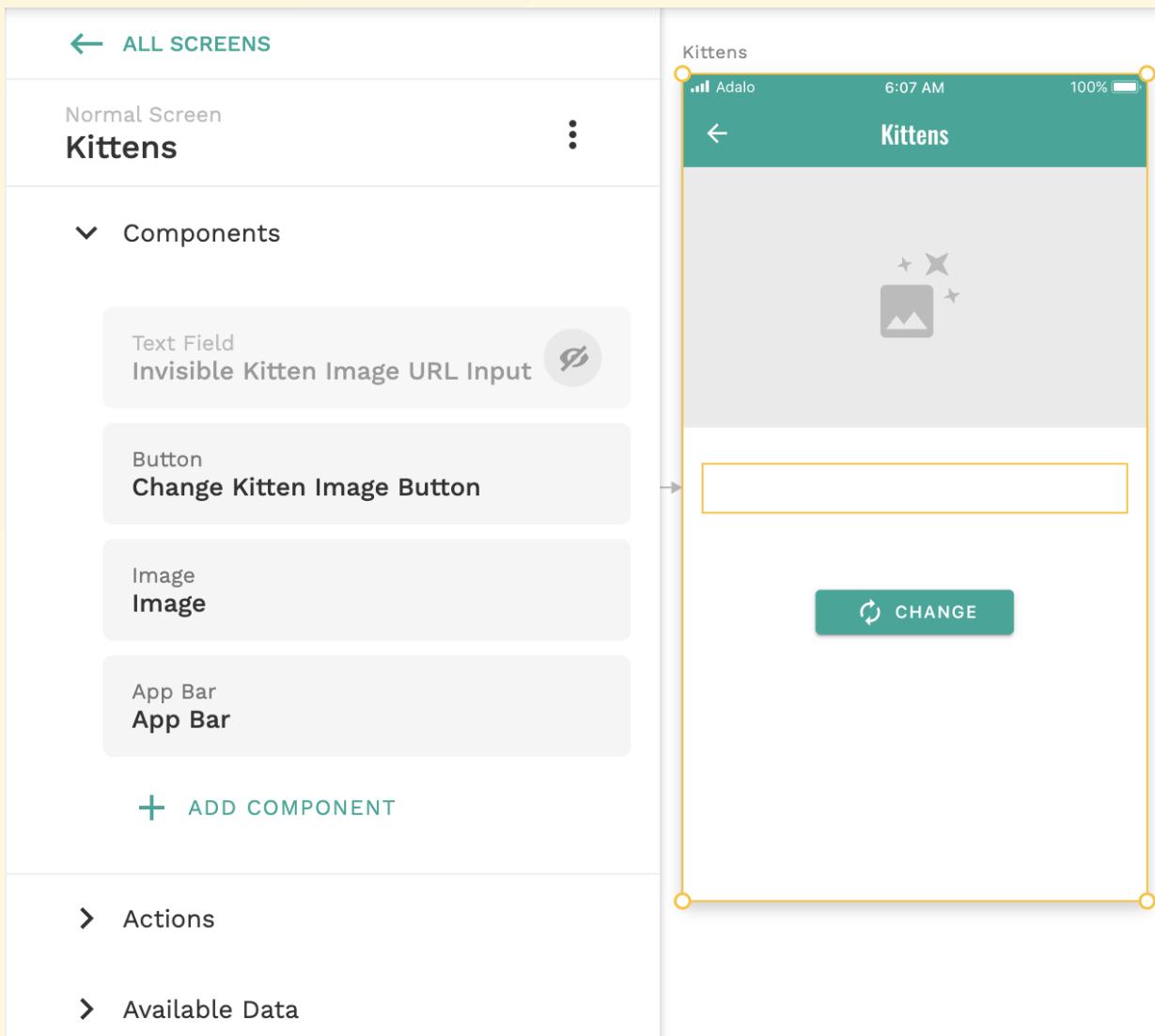
- Set "Invisible Kitten Image URL Input" as Input
- Set Value to "GetRandomKitten > url".
- Click "DONE"



- Click on the Image component
- Set the URL to "Invisible Kitten Image URL Input".

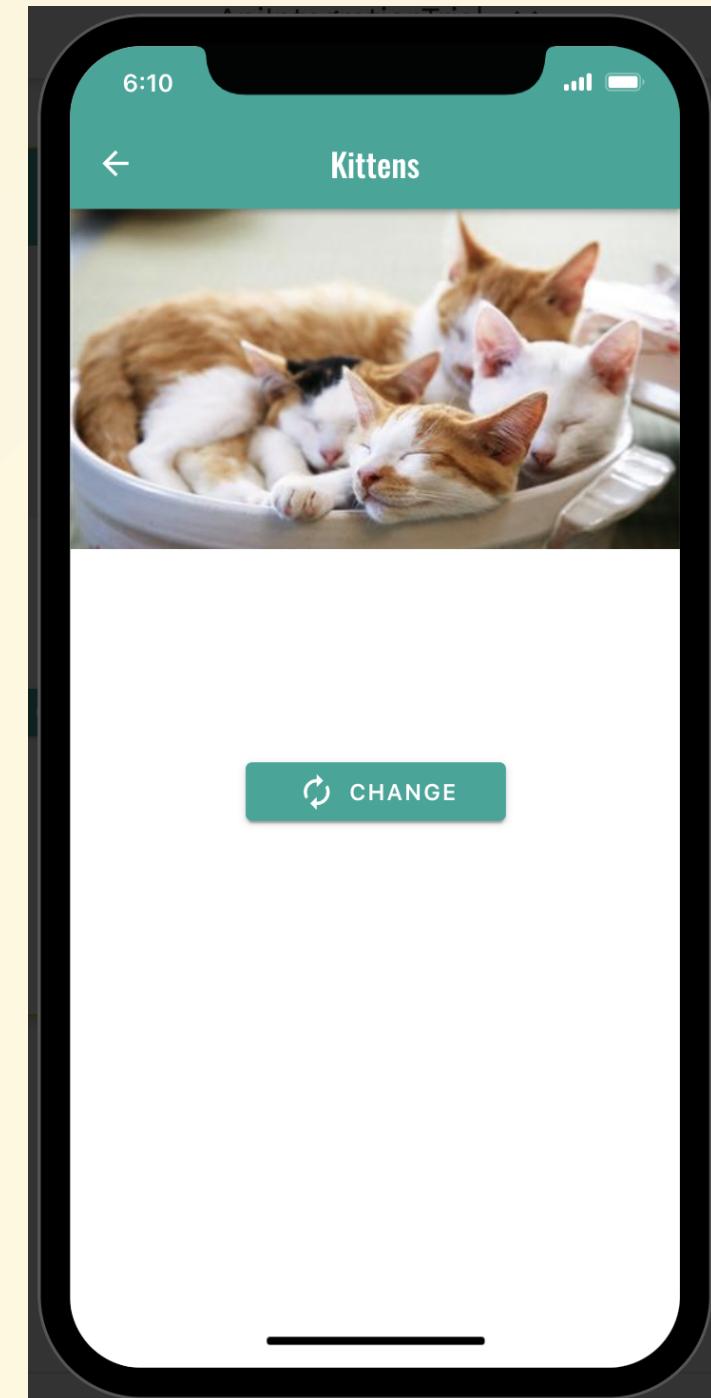


- Click on the screen name "Kittens".
- Click on the eye icon to the right of "Invisible Kitten Image URL Input" to hide it.



Check it with the Preview function.

Click on the CHANGE button and the kitten image is now displayed.



## Supplemental

- Data acquired from the API in Custom Action can be used in subsequent Actions. If you want to use the data in your component, use one of the following methods.
  - Set the data to the Text Input value on the same screen using the Change Input Value of the subsequent Action and load it.
  - Alternatively, you can save the data to the database in a subsequent Action and load it from another component. In this case, it can be accessed even after the screen transition.
    - Example: <https://help.adalo.com/integrations/custom-actions>

## Notes

Currently, Custom Action has some limitations.

- Custom Actions will not work with the Submit button of the Form component.
- If a Custom Action is used as an Action for the entire screen, data obtained as an API response cannot be used in the Imperial Action.
- Cloning an app does not copy the Custom Action; if you clone an app that contains a Custom Action, you will need to recreate it manually.

>Cloning an app does not copy the Custom Action; if you clone an app that contains a Custom Action, you will need to recreate it manually.

The 14-day free trial will end before Development Phase.

If you are likely to use a lot of Custom Actions, we recommend that you create a new Adalo account and start a new free trial before you start working on the Development Phase.

# External Collection

This section describes how to handle data retrieved from the API as an Adalo collection.

If you want to retrieve multiple data at once and list them on the screen, use External Collection instead of Custom Action.

Let's use this API to retrieve and display a list of cat breeds.

<https://docs.thecatapi.com/api-reference/breeds/breeds-list#send-a-test-request>

## 🔗 Send a Test Request

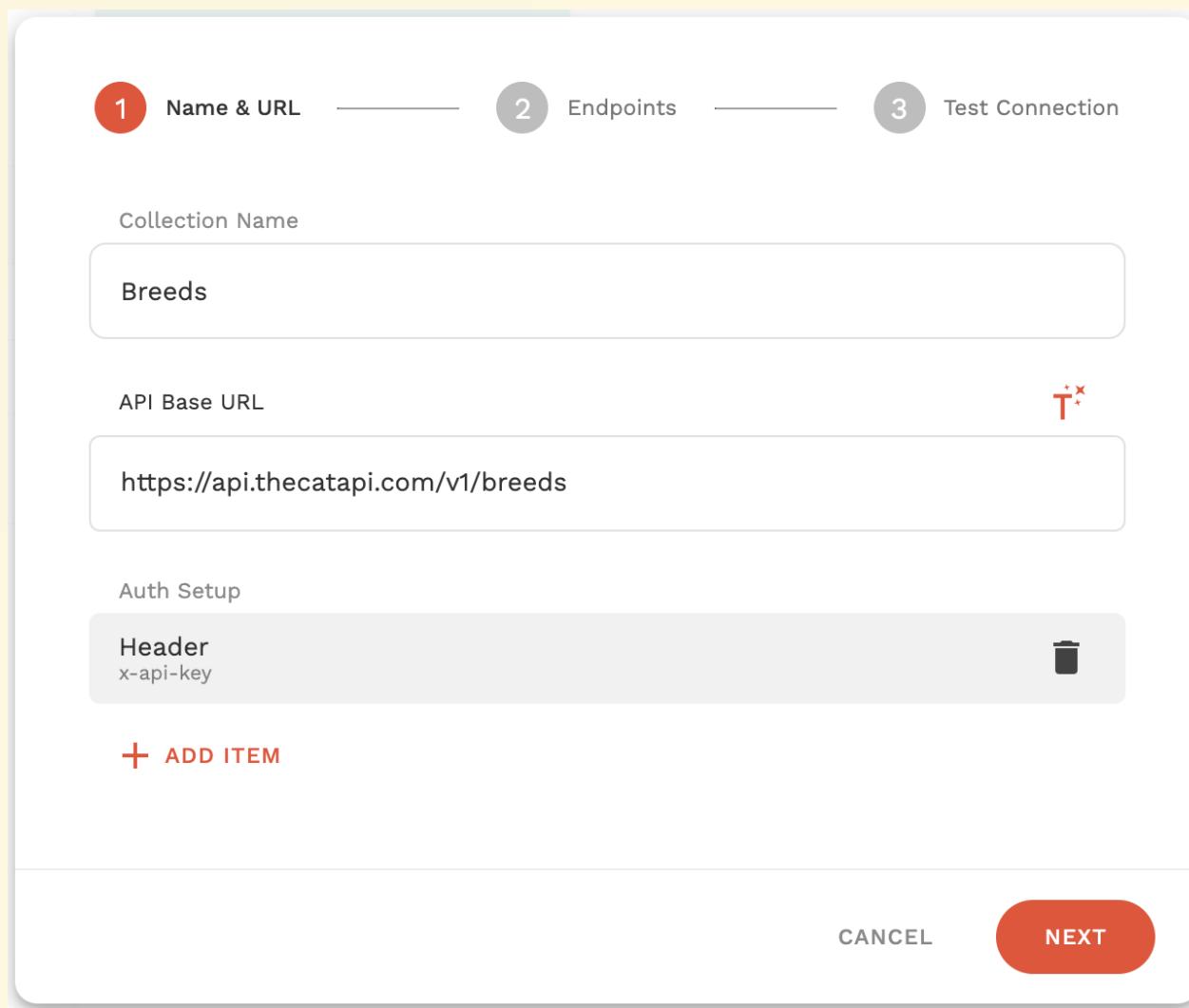
The screenshot shows a user interface for sending a test request to the Cat API. At the top, there is an orange button labeled "Send". To its right, the method "GET" is selected. Below this, the URL "https://api.thecatapi.com/v1/breeds" is displayed. A navigation bar below the URL includes tabs for "Settings", "Headers [1]", "Query [3]", and "Code Generation". The "Headers" tab is currently active. Under the "Headers" tab, there is a button "Add Header" and a key-value pair "x-api-key : {\$\$.env.x-api-key}".

- Click "ADD COLLECTION" in External Collections of Database

The screenshot shows a user interface for managing database collections. On the left, there is a vertical sidebar with several icons: a plus sign inside a circle, a palette, a clipboard, a gear, a checkmark, and a double arrow. The main area is divided into sections:

- Database Collections**: Shows a single collection named "Users" with "1 Record".
- External Collections**: Contains a red button labeled "+ ADD COLLECTION".
- Connect to APIs**: A callout box with the text "Make your app even more powerful by connecting to 3rd party APIs".
- Integration Pack Trial | 14 Days Left**: A callout box with the text "Upgrade now to make sure all of your external collections stay up and running." and a red "UPGRADE NOW" button.

- Collection Name: Breeds
- Base URL:  
<https://api.thecatapi.com/v1/breeds>
- Method: GET
- Auth Setup
  - Header x-api-key: the issued API Key



The screenshot shows a mobile application interface for managing API collections. At the top, there are three circular tabs labeled 1 Name & URL, 2 Endpoints, and 3 Test Connection. The first tab is highlighted with a red background.

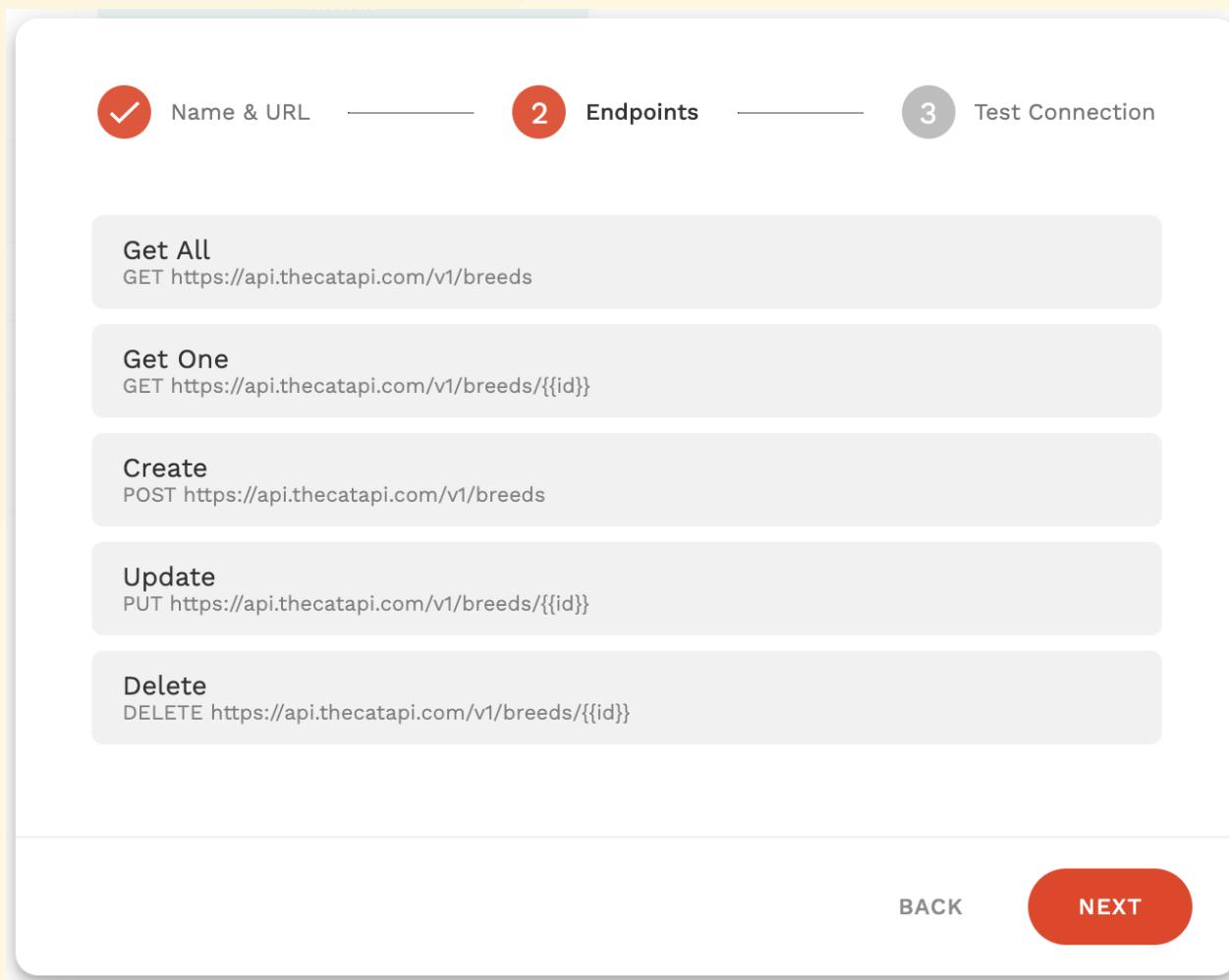
The main area contains the following fields:

- Collection Name:** Breeds
- API Base URL:** https://api.thecatapi.com/v1/breeds
- Auth Setup:** A section containing a single item: **Header** x-api-key. To the right of this item is a small trash can icon.

At the bottom right of the screen are two buttons: **CANCEL** and a large **NEXT** button.

In Adalo, you can set up five Endpoints (access methods) for each resource (in this example, breeds) that you want to access with the API.

Depending on the specifications of the API, you may need to modify it to fit your needs, but in this case, you can just click NEXT.



- Run the test and if it succeeds, click "CREATE COLLECTION".

The screenshot shows a user interface for testing a MongoDB API endpoint. At the top, there are three status indicators: "Name & URL" (green checkmark), "Endpoints" (green checkmark), and "Test Connection" (red circle with the number 3). Below these, the text "Test Successful!" is displayed in bold, followed by the message "Here's the full response we got back:". A JSON response is shown, representing a single document for the Abyssinian cat breed. The JSON data includes various fields such as weight ranges, URLs for different sources, and various temperament and care-related metrics.

```
[
  {
    "weight": {
      "imperial": "7 - 10",
      "metric": "3 - 5"
    },
    "id": "abys",
    "name": "Abyssinian",
    "cfa_url": "http://cfa.org/Breeds/BreedsAB/Abyssinian.aspx",
    "vetstreet_url": "http://www.vetstreet.com/cats/abyssinian",
    "vcahospitals_url": "https://vcahospitals.com/know-your-pet/cat-breeds/",
    "temperament": "Active, Energetic, Independent, Intelligent, Gentle",
    "origin": "Egypt",
    "country_codes": "EG",
    "country_code": "EG",
    "description": "The Abyssinian is easy to care for, and a joy to have in your home. They are intelligent, active, and affectionate cats that bond well with their families. Their distinctive coat pattern and unique personality make them popular pets.", "life_span": "14 - 15",
    "indoor": 0,
    "lap": 1,
    "alt_names": "",
    "adaptability": 5,
    "affection_level": 5,
    "child_friendly": 3,
    "dog_friendly": 4,
    "energy_level": 5,
    "grooming": 1,
    "health_issues": 2,
    "intelligence": 5,
    "shedding_level": 2,
    "social_needs": 5,
    "stranger_friendly": 5,
    "vocalisation": 1,
    "experimental": 0
  }
]
```

**BACK** **CREATE COLLECTION**

The External Collection has been created.

All the data to be retrieved from the API is set as properties.

External Collections

▼  Breeds 

A id

A weight > imperial

A weight > metric

A name

A temperament

A origin

A country\_codes

A country\_code

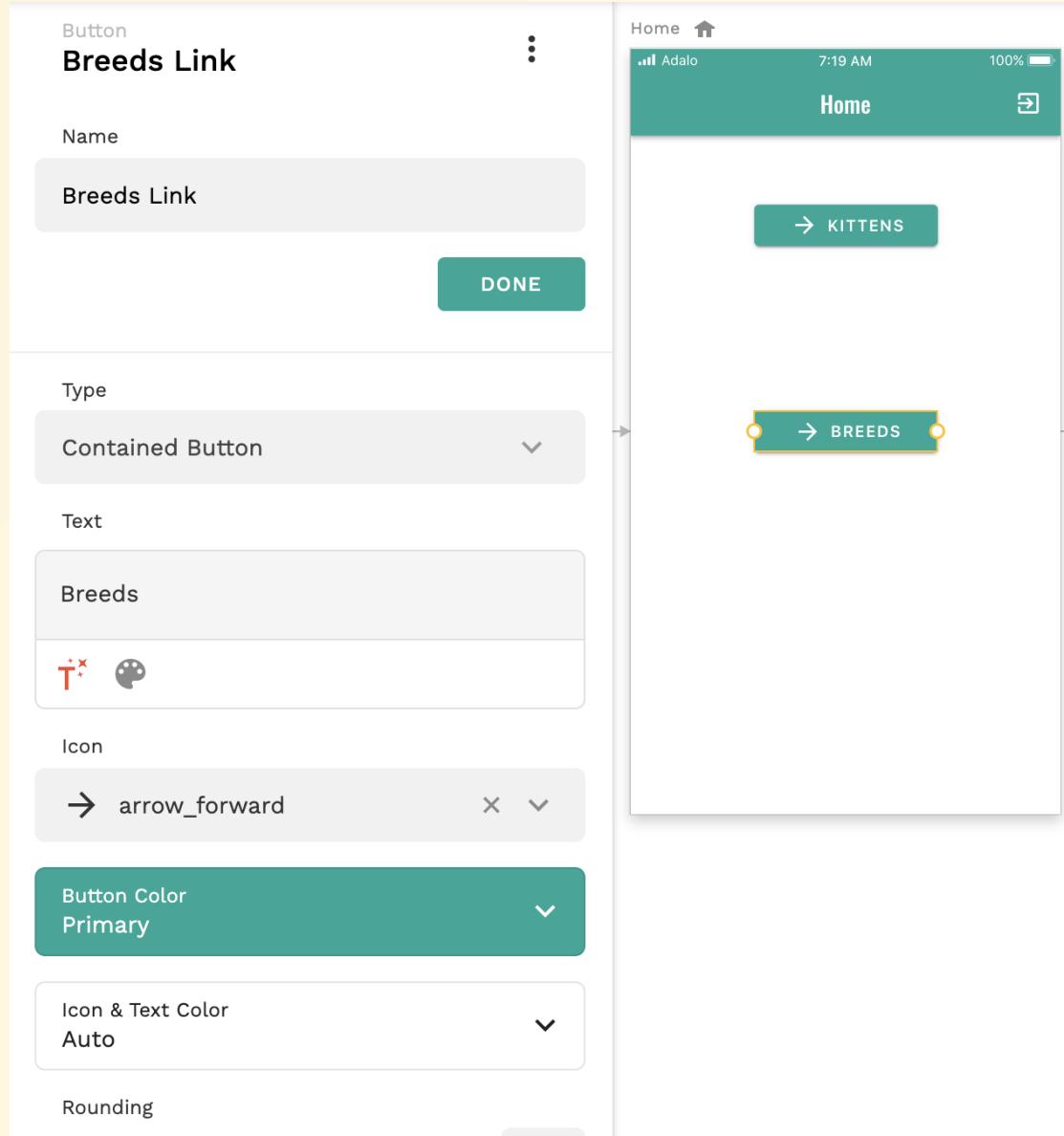
A description

A life\_span

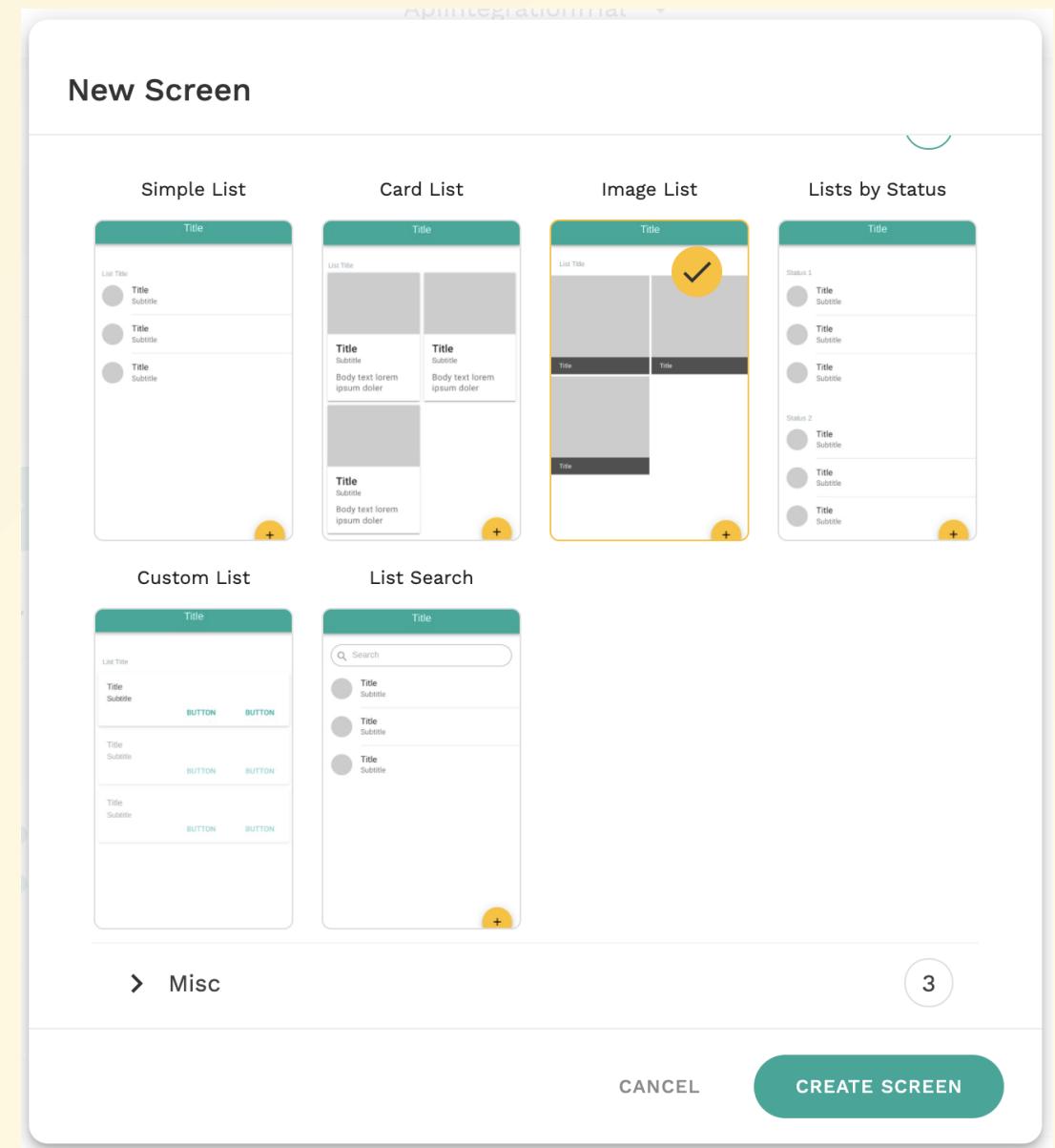
 indoor

# Let's list the retrieved data.

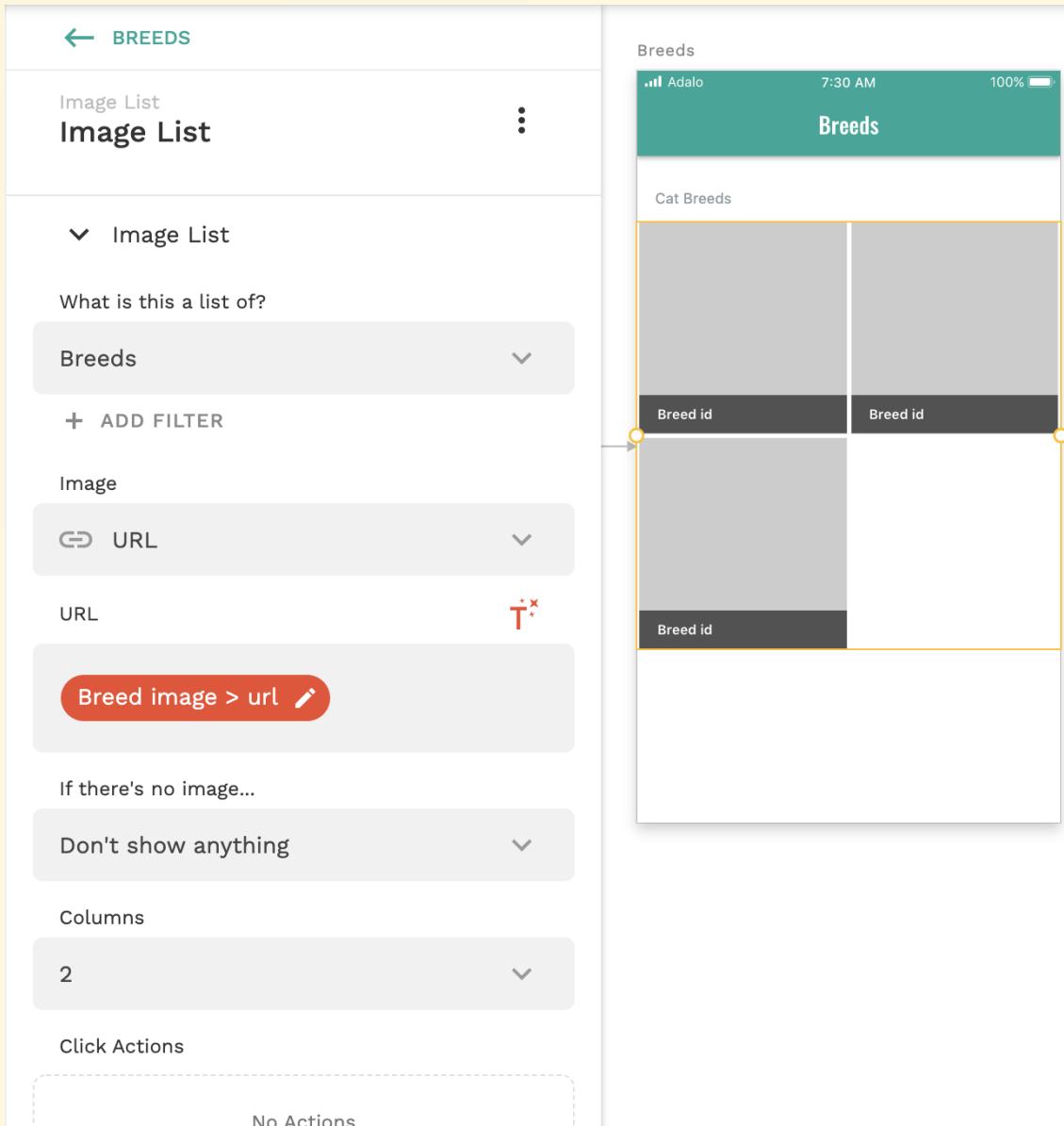
- Add a "Breeds Link" button to the Home screen.
- Add a LINK to the NEW SCREEN from ADD ACTION.



- Enter "Breeds" in Name.
- Select Image List in Template
- Click CREATE SCREEN



- Change the List Title Text to "Cat Breeds".
- Set the Image List as the Breeds Collection list.
- Set Image URL to "image > url".
- "If there's no image..." Set "Don't show anything" to "If there's no image...".
- Remove the + button at the bottom right.



- Change Text to "name".

The image shows a mobile application interface with two main sections: a configuration screen on the left and a preview screen on the right.

**Configuration Screen (Left):**

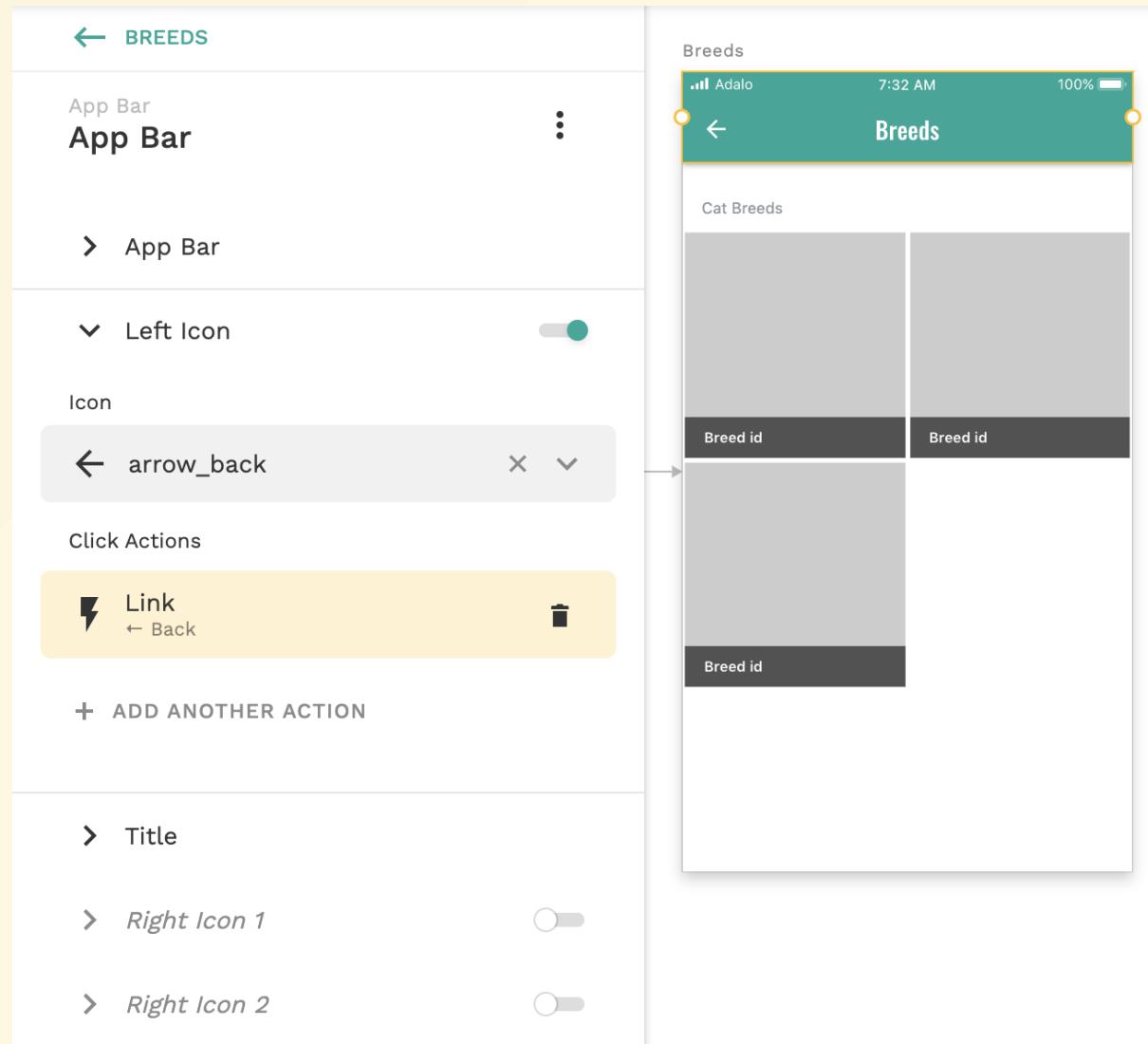
- List Header:** A toggle switch is turned off.
- Text:** A toggle switch is turned on.
- Title:** A red input field labeled "Breed name" with a pencil icon.
- Subtitle:** An input field with a placeholder "Enter Text..." and a red T icon.
- Subtitle Position:** A dropdown menu with the placeholder "Select...".
- Text Position:** A dropdown menu with the placeholder "Select...".
- Background Effect:** A dropdown menu with the placeholder "Select...".
- Image Style:** A section indicated by a right-pointing arrow.

**Preview Screen (Right):**

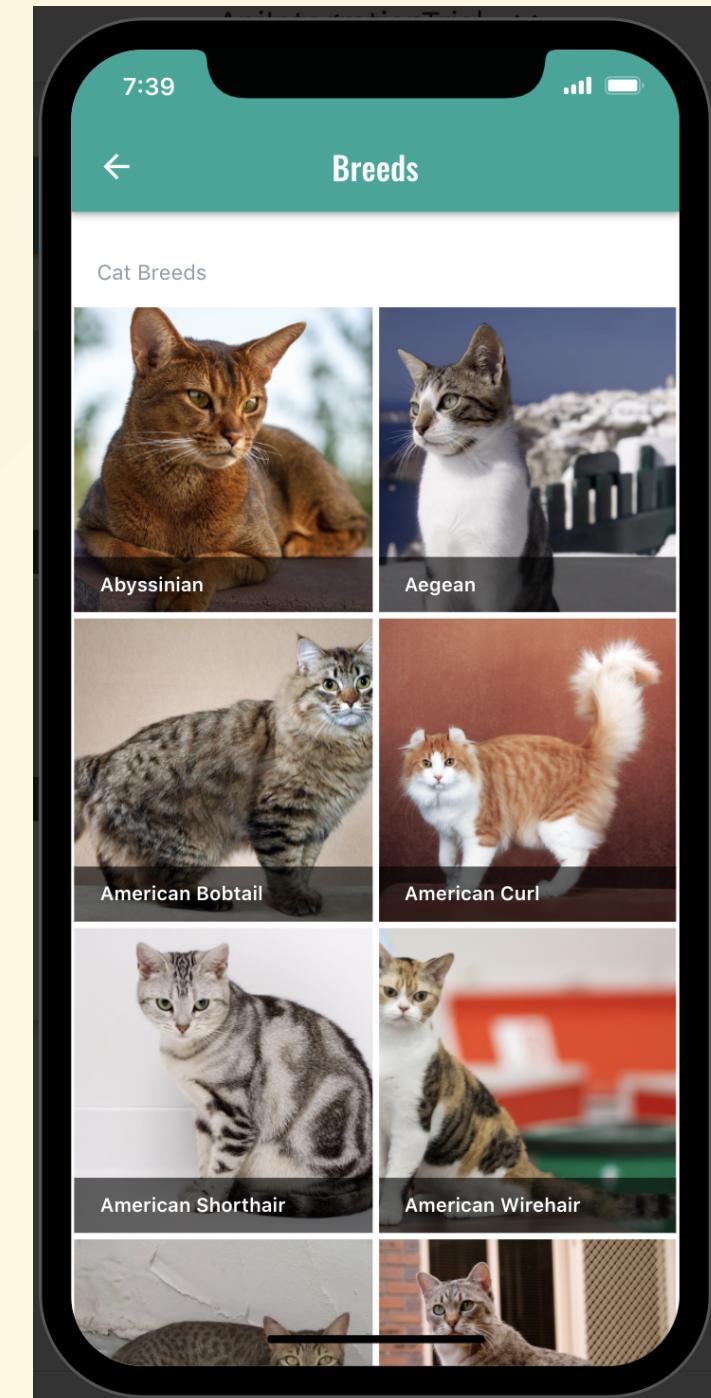
- Breeds:** The title bar shows "Adalo" and "7:37 AM" with a 100% battery icon.
- Cat Breeds:** The section title.
- Table:** A grid table with three columns and four rows. The first row is a header with "Breed name" in each column. The second row has "Breed name" in the first column and "Breed name" in the third column. The third row has "Breed name" in the first column and is highlighted with a yellow border. The fourth row is empty.

Show the back icon since it is hidden.

- Click on the App Bar
- Change the Left Icon toggle to ON.



Checking with the Preview function, the list of cat breeds is now displayed.

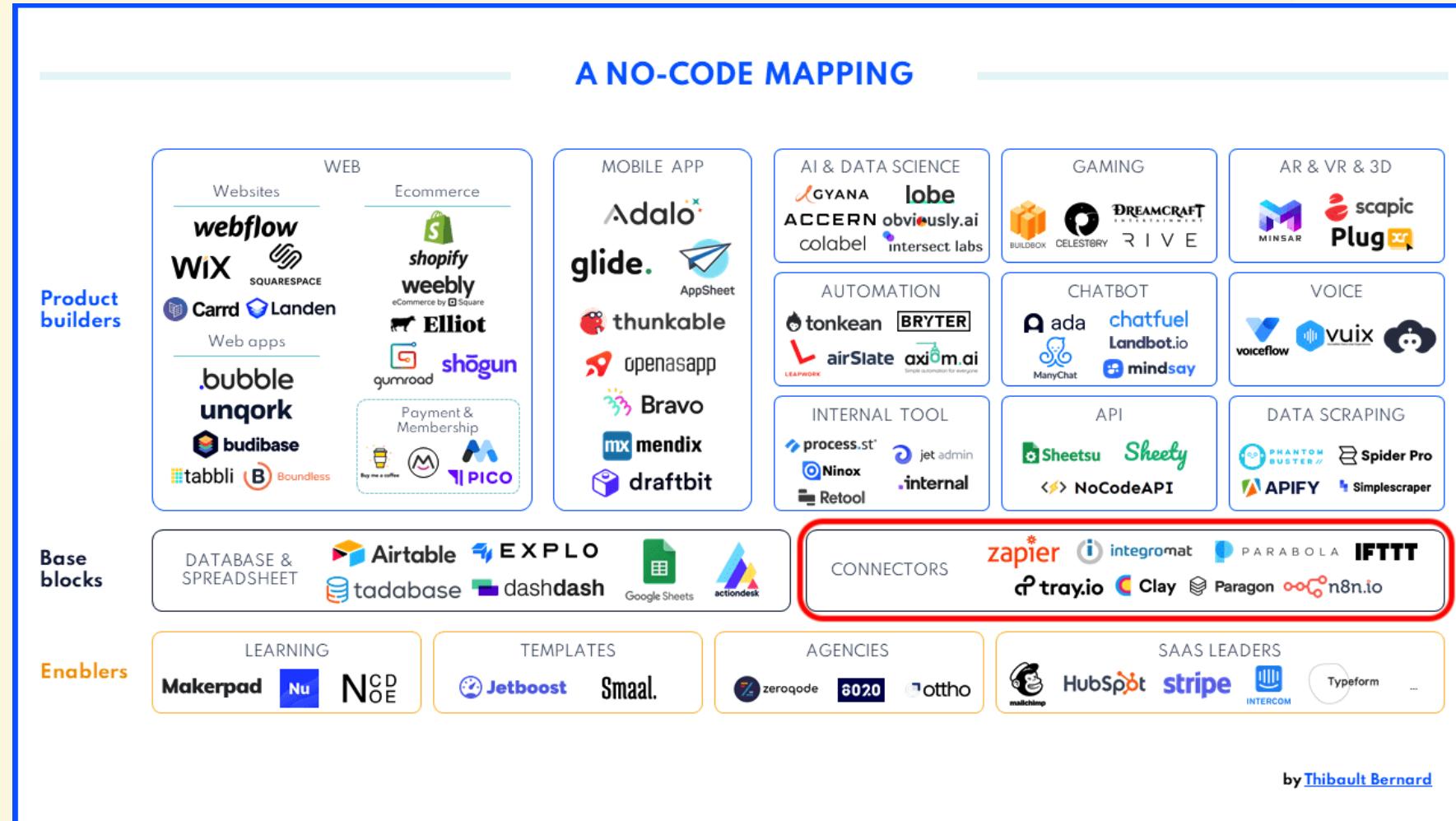


## Connector Service

It is required to understand api document to use Custom Action and External Collection, but it may be difficult.

Next, I will introduce connector service that can be more easily used to integrate adalo app with external services.

There are some connector services, but this time we will use a service called Zapier.



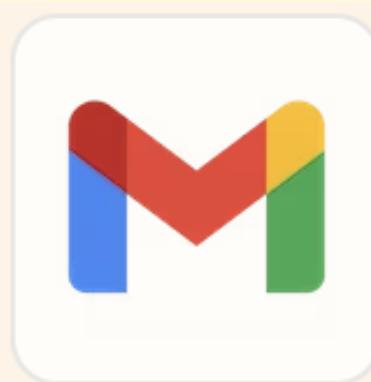
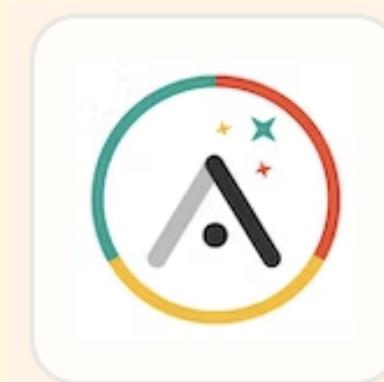
# Zapier

- Using Zapier, you can easily integrate services by following the instructions. Let's give it a try.

The screenshot shows the Zapier website with a specific integration for Adalo. At the top, there's a navigation bar with links for Product, Solutions, Resources & Support, Pricing, Contact Sales, Log in, and a prominent orange "Sign up" button. Below the navigation, a teal sidebar on the left features the Adalo logo and the text "Adalo Integrations App Builder". The main content area has tabs for "Integrations" (which is selected) and "Help". A section titled "Do more with Adalo integrations" explains that Zapier lets you connect Adalo with thousands of other apps to automate work. It includes a large orange button labeled "Connect Adalo to 5,000+ apps". Below this, a horizontal line of logos for various companies (Slack, Meta, Shopify, Dropbox, Asana) indicates their trust in Zapier. At the bottom, there's a search bar with the placeholder "Search for another app...".

Adalo does not have an email sending function.

This time, we will use Zapier to integrate Adalo with Gmail so that Welcome email will be automatically sent to people who SignUp to the application.



- Create a Google account, as you will need it to use Gmail.
  - If you already have one, you can use that account, so you do not need to create new one.
  - If you do not have one, please create one with me. Click following URL.

[https://accounts.google.com/signup/v2/webcreateaccount?  
continue=https%3A%2F%2Faccounts.google.com%2FManageAc  
count%3Fnc%3D1&dsh=S50453738%3A1637917137418951&  
biz=false&flowName=GlifWebSignIn&flowEntry=SignUp](https://accounts.google.com/signup/v2/webcreateaccount?continue=https%3A%2F%2Faccounts.google.com%2FManageAccount%3Fnc%3D1&dsh=S50453738%3A1637917137418951&biz=false&flowName=GlifWebSignIn&flowEntry=SignUp)

- Once you have your Google account ready, go to <https://zapier.com/apps/adalo/integrations> and click on "Connect Adalo to 5,000+ apps".

The screenshot shows the Zapier website with a specific focus on the Adalo Integrations page. At the top, there's a navigation bar with links for Product, Solutions, Resources & Support, Pricing, Contact Sales, Log in, and a prominent orange Sign up button. Below the navigation, there's a teal sidebar featuring the Adalo logo (a stylized 'A' inside a circle) and the text 'Adalo Integrations App Builder'. The main content area has a light beige background. It features a sub-navigation bar with 'Integrations' (which is underlined in blue) and 'Help'. Below this, the heading 'Do more with Adalo integrations' is displayed in bold black text. A descriptive paragraph follows, stating: 'Zapier lets you connect Adalo with thousands of the most popular apps, so you can automate your work and have more time for what matters most—no code required.' At the bottom of the main content area is a large orange button with white text that reads 'Connect Adalo to 5,000+ apps'. At the very bottom of the page, a horizontal line leads to the footer text: 'Top companies trust Zapier to automate work and free up time'.

- Sign up by any method you like.

The screenshot shows the Zapier sign-up page. At the top right, there are links for "Contact Sales" and "Log in". The main heading reads: "Join millions worldwide who automate their work using Zapier." Below this are three social login buttons: "Sign up with Google" (blue), "Sign up with Facebook" (dark blue), and "Sign up with Microsoft" (black). A horizontal line with the word "OR" follows. Below this, there is a "Work Email (required)" field with a placeholder "Work Email". To the right, there are two side-by-side fields for "First Name (required)" and "Last Name (required)", each with a placeholder "First Name" and "Last Name" respectively. At the bottom center is a large orange button labeled "Get Started Free". At the very bottom, a small note states: "By signing up, you agree to Zapier's [terms of service](#) and [privacy policy](#)".

**-zapier**

Contact Sales Log in

Join millions worldwide who automate their work using Zapier.

- ✓ Easy setup, no coding required
- ✓ Free forever for core features
- ✓ 14-day trial of premium features & apps

Sign up with Google

Sign up with Facebook

Sign up with Microsoft

OR

Work Email (required)

Work Email

First Name (required)

Last Name (required)

First Name

Last Name

Get Started Free

By signing up, you agree to Zapier's [terms of service](#) and [privacy policy](#).

- Enter roles and number of employees and click "Continue".

Hey Ryo, help us customize your experience by  
telling us a bit about yourself

Select the role that best describes you (required)

IT

Data Science

Customer Support Ops

Sales Ops

Project Management

HR & Recruiting Ops

Marketing & Marketing Ops

Engineering

Business Owner

Other

How many employees does your company have? (required)

1-49

50-249

250-999

1,000+

Continue

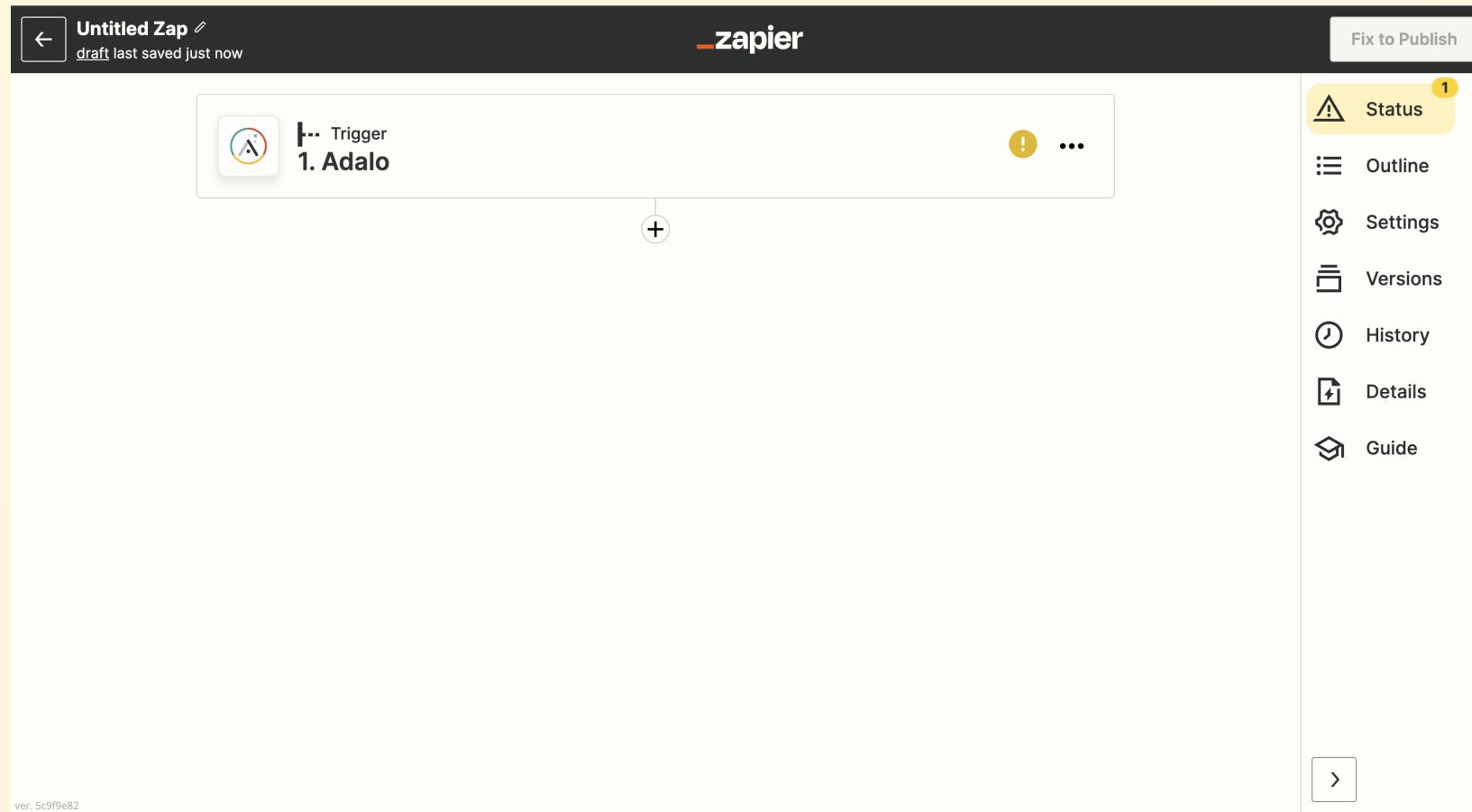
- Add Adalo and Gmail to the list of applications to use and click "Finish setup".

The image displays two side-by-side screenshots of the Adalo app setup process, specifically the step where users select the apps they use.

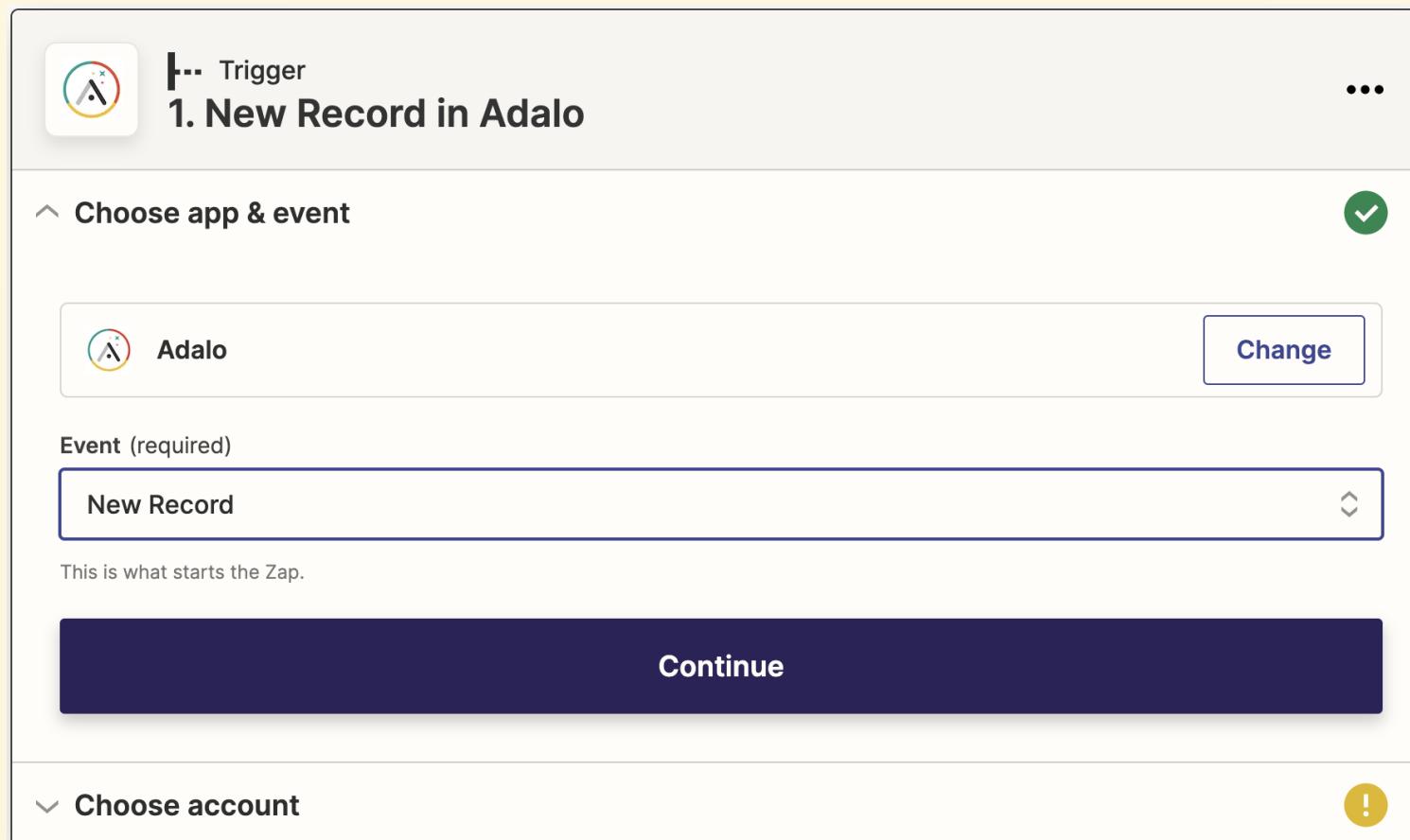
**Left Screenshot:** The title is "What apps do you use?". Below it, a subtitle says "We'll give you personalized workflow recommendations based on the apps you choose." A search bar contains the text "adalo". Below the search bar is a list containing one item: "Adalo" with its icon. At the bottom are two circular buttons: "Gmail X" and "Adalo X". A progress bar at the bottom indicates "3 apps away from tailored recommendations". A "Skip" button is on the left, and a prominent blue "Finish setup" button is at the bottom right.

**Right Screenshot:** The title is "What apps do you use?". Below it, a subtitle says "We'll give you personalized workflow recommendations based on the apps you choose." A search bar contains the text "gmail". Below the search bar is a list containing three items: "Gmail" (with its icon), "Sortd for Gmail" (with its icon), and "BigMailer" (with its icon). At the bottom are two circular buttons: "Adalo X" and "Gmail X". A progress bar at the bottom indicates "3 apps away from tailored recommendations". A "Skip" button is on the left, and a prominent blue "Finish setup" button is at the bottom right.

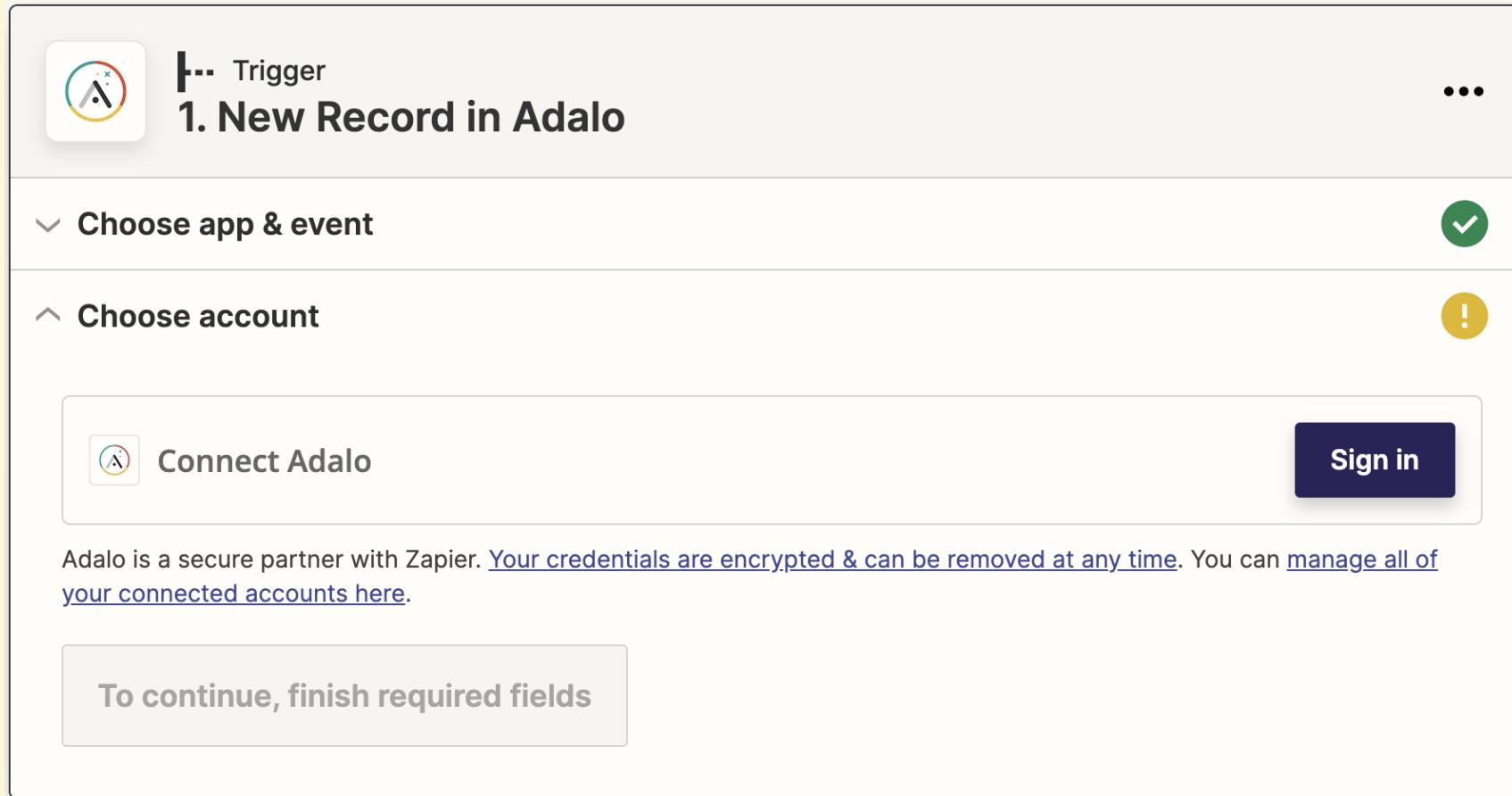
You will see an edit screen for service integration settings, called Zap.



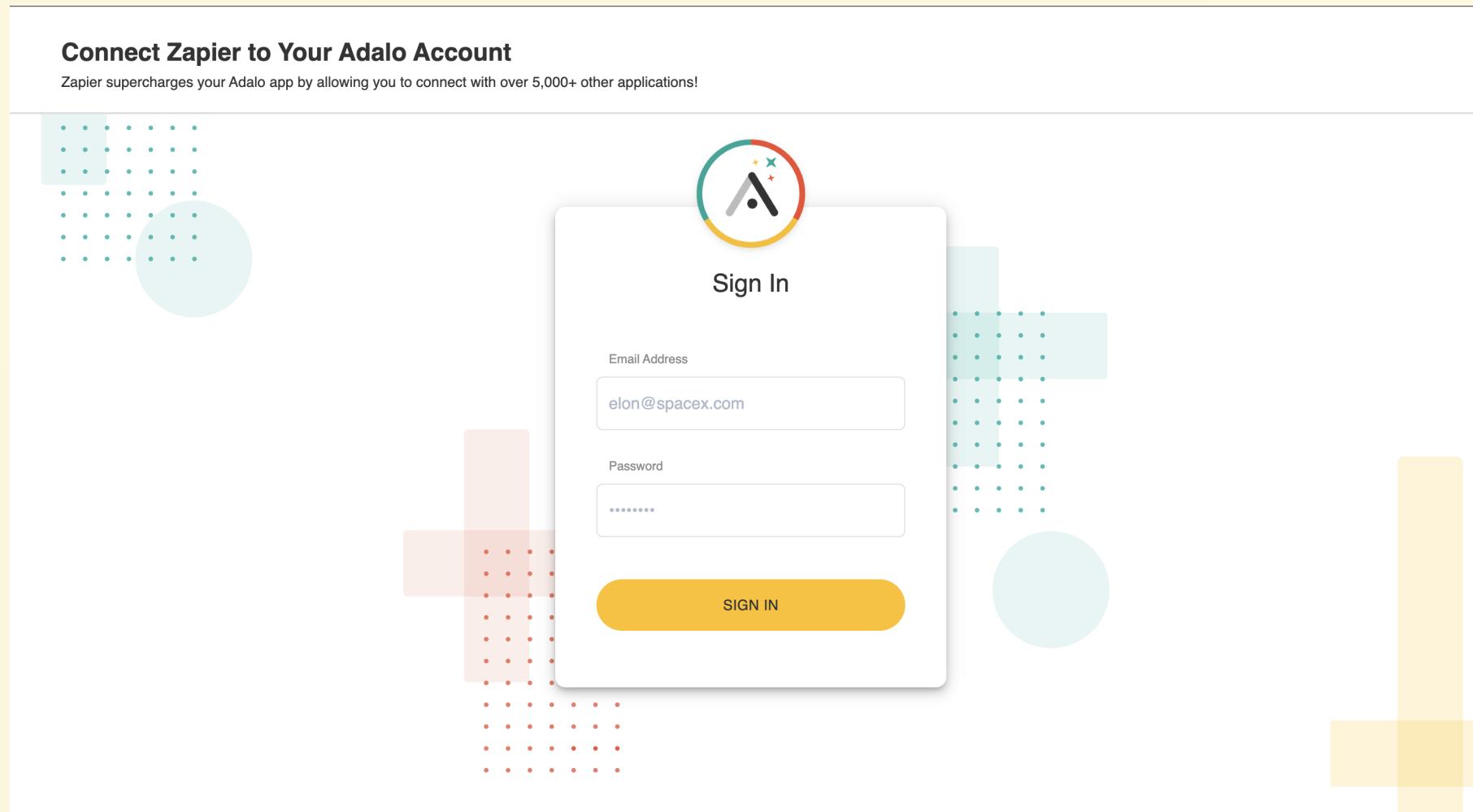
- Click on Trigger (= rectangle containing "1. Adalo")
- Set "New Record" as Event
- Click "Continue"



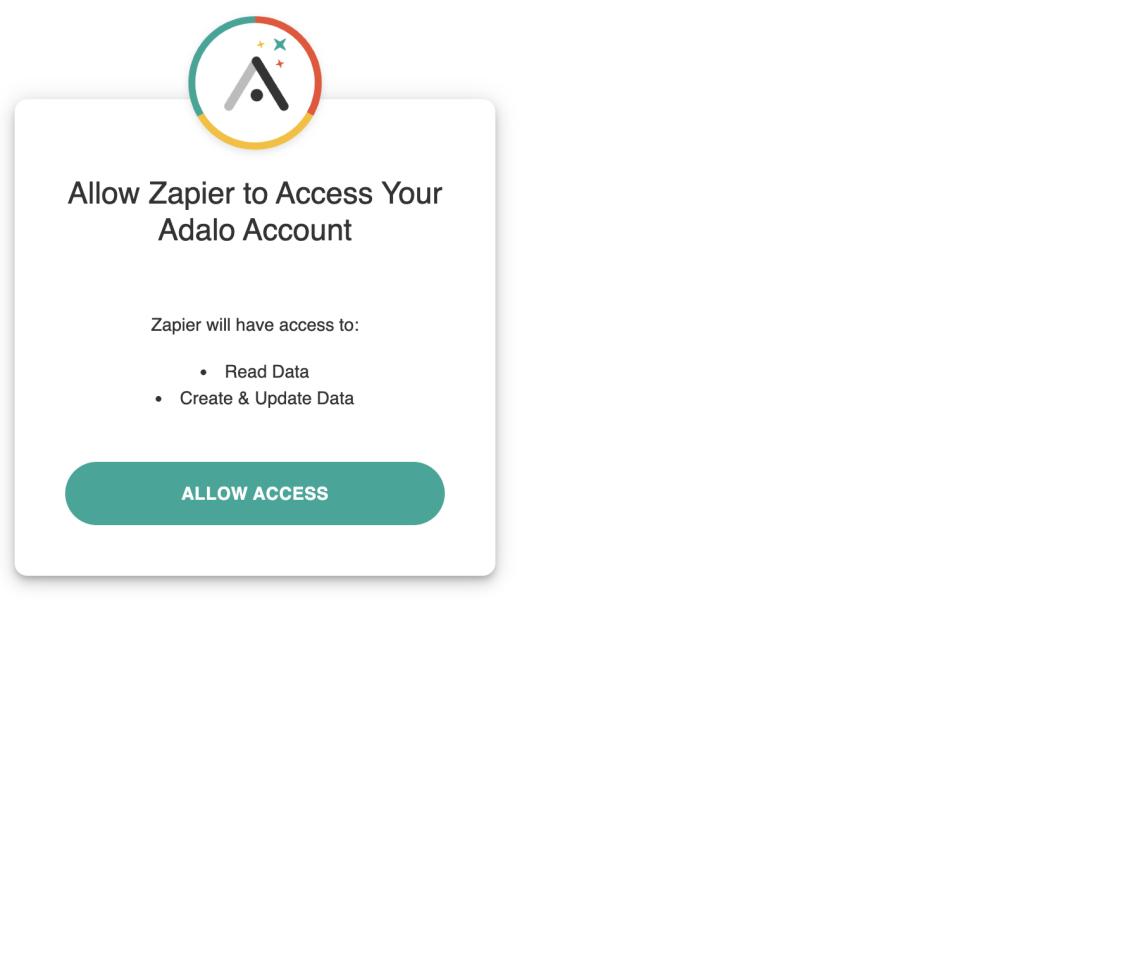
- Click "Sign in"



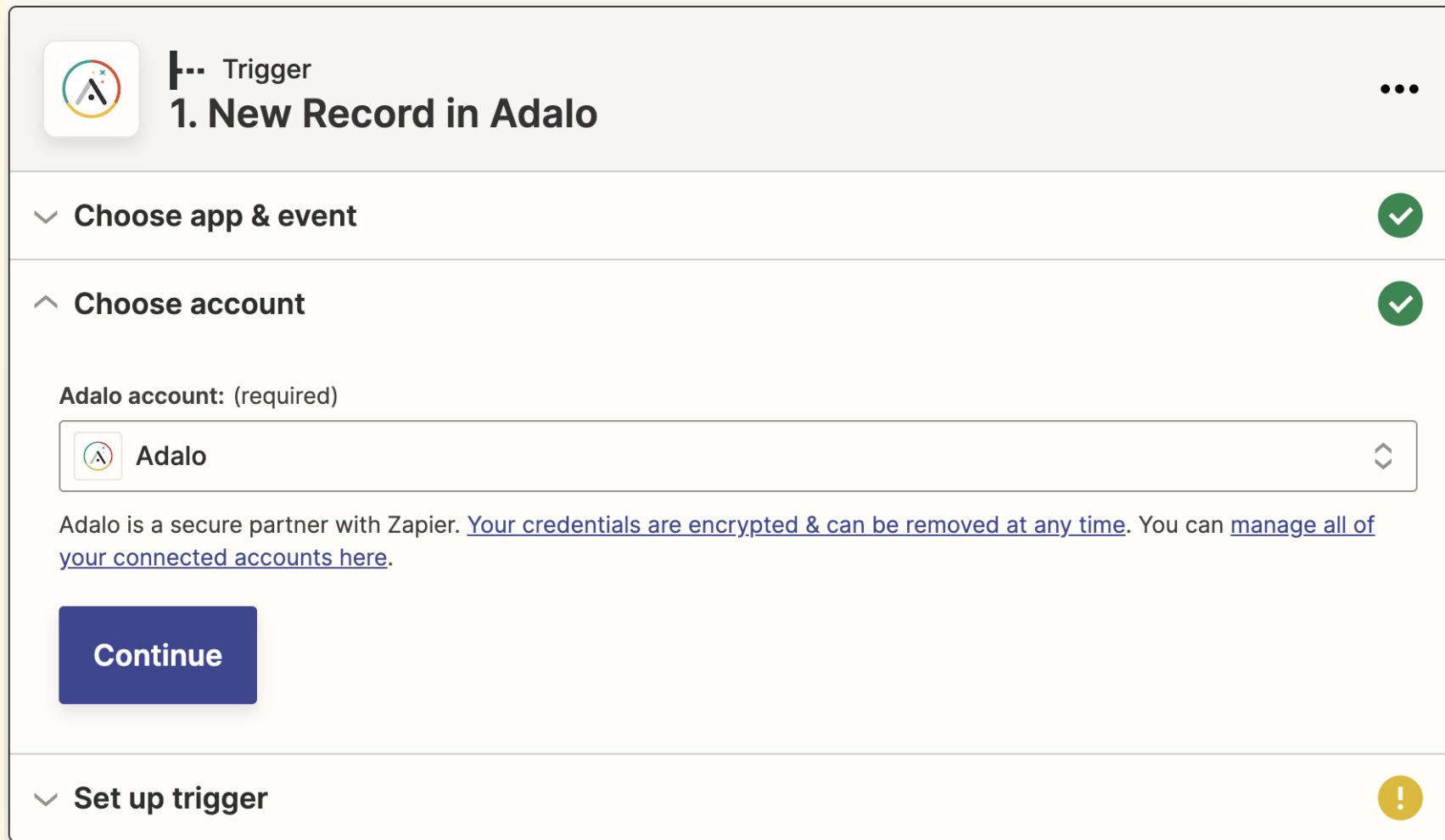
- Enter the Email Address and Password of your Adalo account and click "SIGN IN"



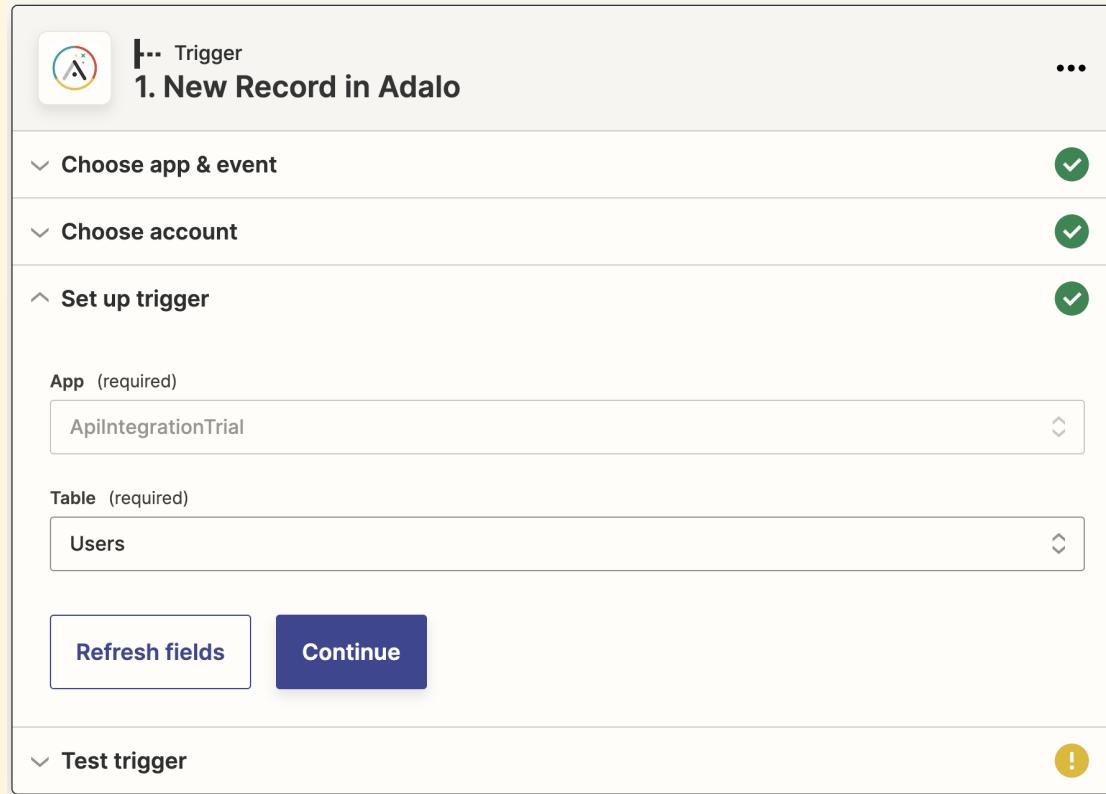
- Click "ALLOW ACCESS"



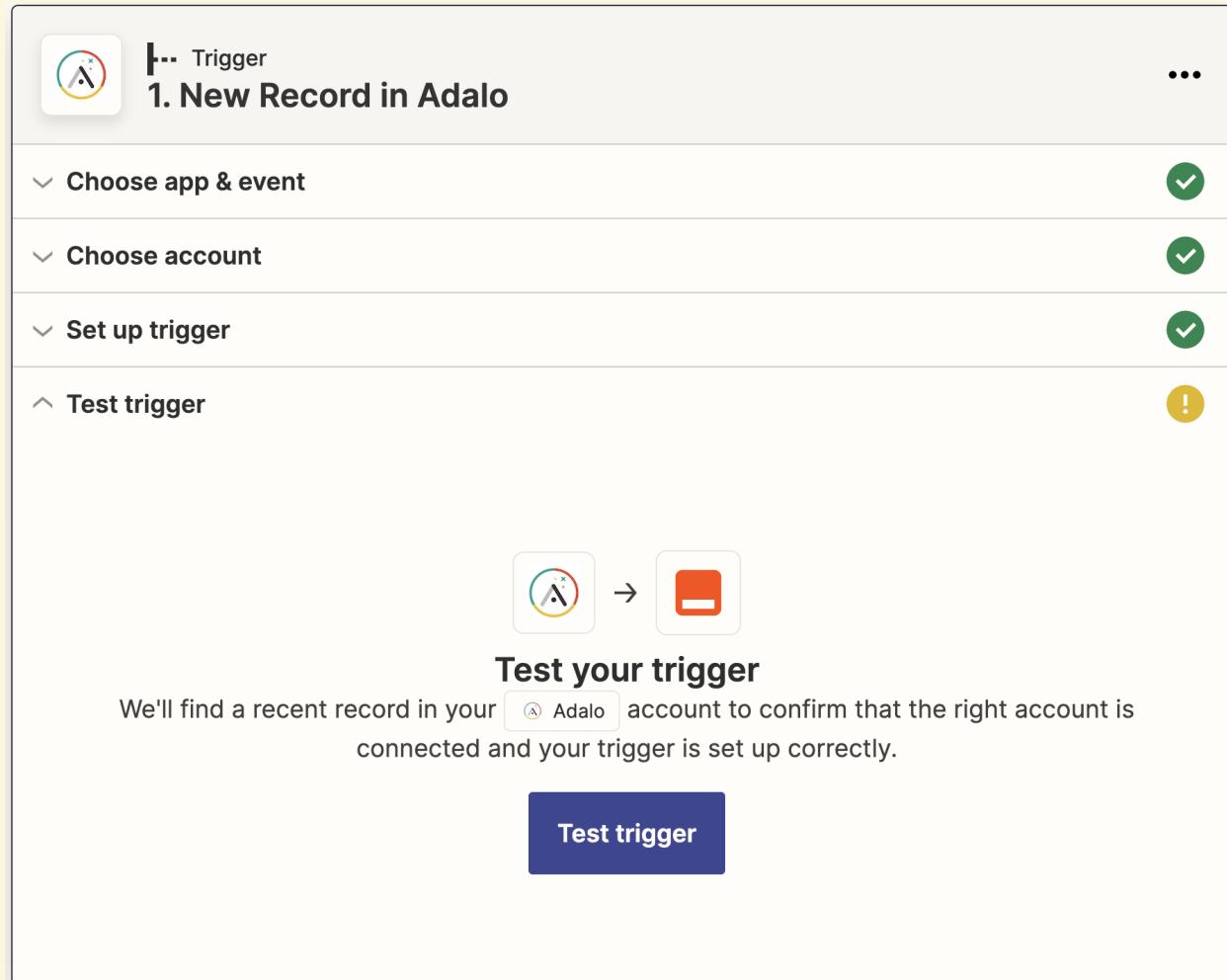
- Confirm that your Adalo account has been set up and click "Continue"



- Set "ApilntegrationTrial" in App.
- Set "Users" in Table
- Click "Continue"



- Click "Test trigger"



- Click "Continue" after Zapier found a record

The screenshot shows the Zapier test trigger interface. At the top left is a collapse arrow labeled "Test trigger". In the top right corner is a green circular button with a white checkmark. Below the header, a green circular icon with a white checkmark is followed by the text "We found a record!". A subtext below it reads "This record was found in your Adalo account. [Learn more about test data.](#)". A dropdown menu is open, showing "Record A" with a small Adalo logo icon. Below the dropdown is a search bar with a magnifying glass icon. The main content area displays the record details:  
id: 1  
Email: imahashi@example.com  
Password: [hidden]  
Username: null  
Full Name: Ryo Imahashi  
Temporary Password: null  
Temporary Password Expira... : null

A large blue "Continue" button is located at the bottom of the interface.

- Select Gmail as the App to execute the Action.

The screenshot shows the Zapier interface for selecting an action app. At the top, it says "2. Action" and defines an action as "An action is an event a Zap performs after it starts". There are "Learn more" and "..." buttons. Below this, there are two main sections: "App event" and "Built-in tools".

**App event:** "Do something in an app". It includes a search bar labeled "Search apps..." and a grid of app icons:

- Adalo
- Gmail
- Google Sheets
- Filter by Zapier
- Slack
- Google Calendar
- Webhooks by Zapier  
Premium
- Formatter by Zapier
- Email by Zapier
- Mailchimp

...and over 5,200+ more

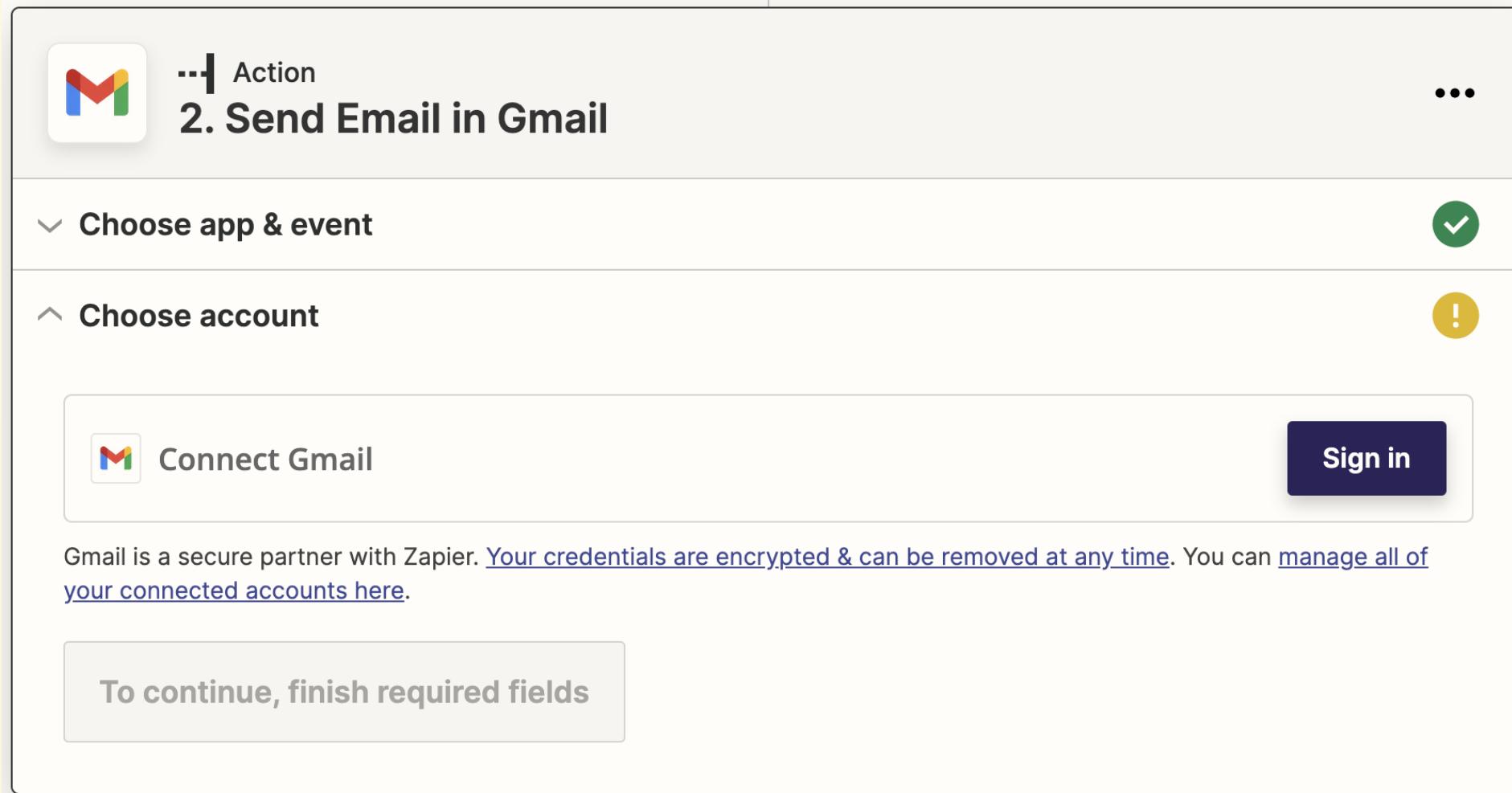
**Built-in tools:** "Try one of our [20+ built-in tools](#)". It lists four tools:

- Path**: Build different steps for different rules.
- Delay**: Pause actions for a certain amount of time.
- Filter**: Only proceed when a condition is met.
- Format**: Change how incoming data is formatted.

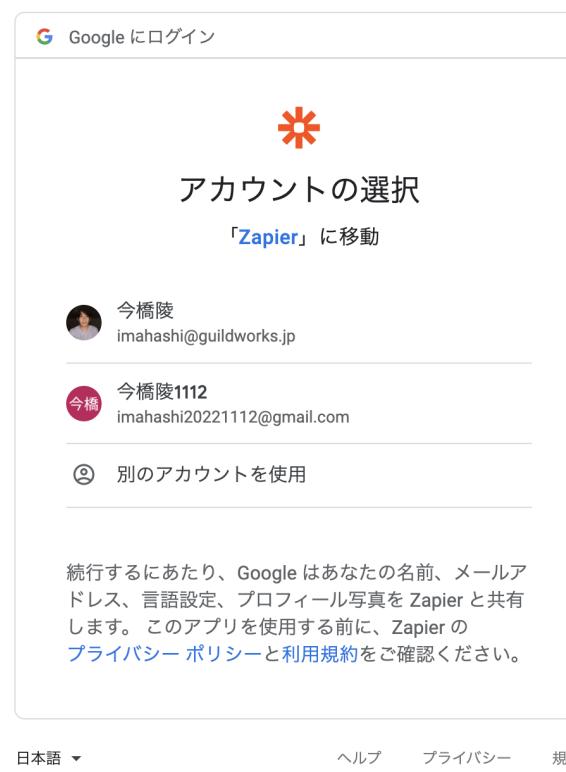
- Set "Send Email" to "Event".
- Click "Continue"

The screenshot shows the configuration interface for a 'Send Email in Gmail' action in Zapier. At the top, there's a header with the Zap logo, the text 'Action', and '2. Send Email in Gmail'. Below this, a section titled 'Choose app & event' has a 'Gmail' icon and a 'Change' button. Under 'Event (required)', a dropdown menu is open, showing 'Send Email' as the selected option. A note below the dropdown states: 'This is performed when the Zap runs.' At the bottom of the screen is a large blue 'Continue' button. At the very bottom, there's a partially visible section titled 'Choose account' with a yellow exclamation mark icon.

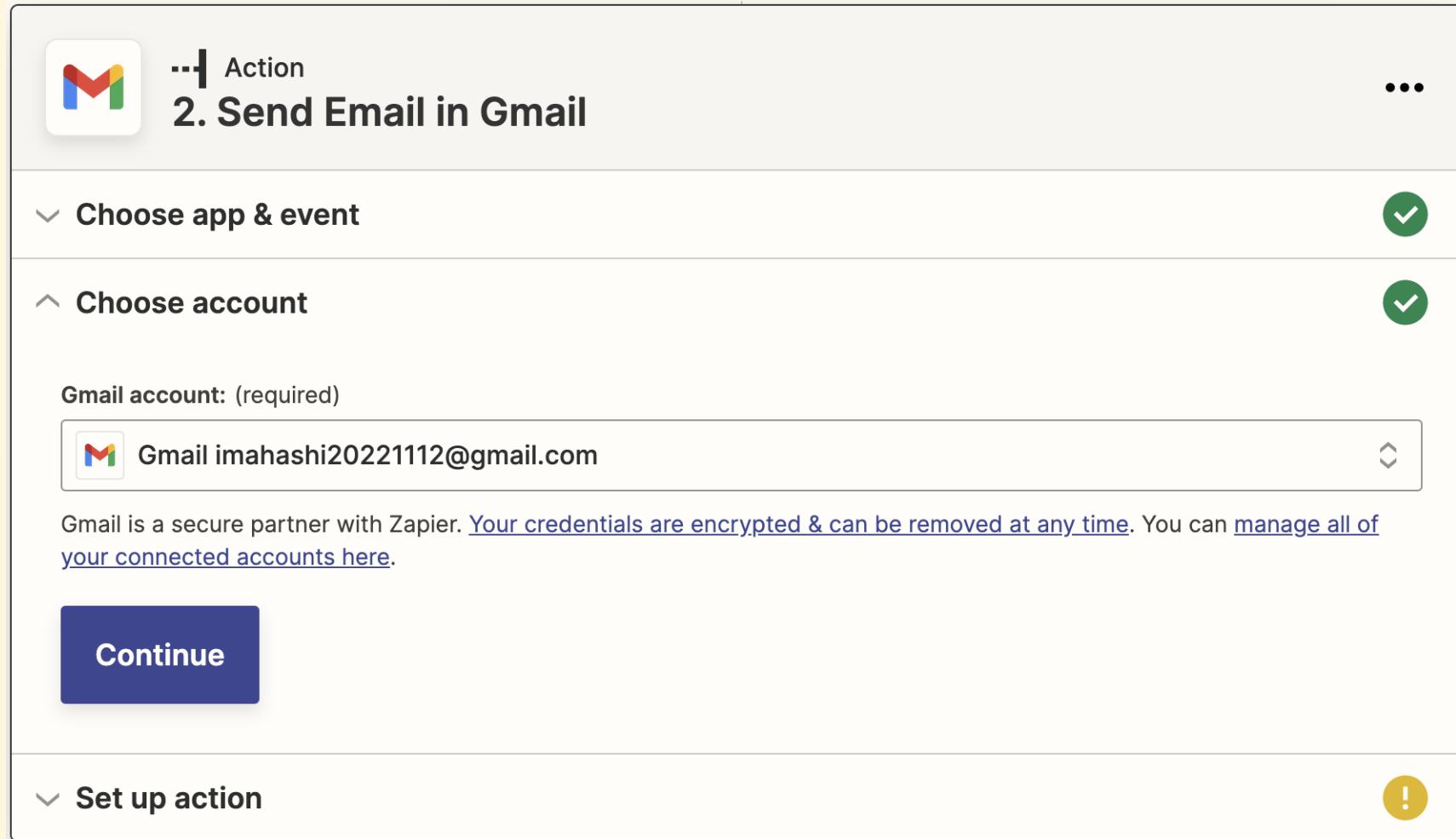
- Click "Sign in"



- Select the account which sends Welcome email. (Select account of From Address)



- Confirm that the Gmail account is set, and click "Continue".



- Set Email in To
- Set your Gmail address in From
- Set your app's name in From Name
- Enter Subject and Body as you like
- Click "Continue"

To

 1. Email: imahashi@example.com X

Enter text or insert data...

Who will this email be sent to? Multiple email addresses can be entered either individually, or as a comma separated... [more](#)

Cc

Enter text or insert data...

Who should be cc'd on this email? Multiple email addresses can be entered either individually, or as a comma separat... [more](#)

Bcc

Enter text or insert data...

Who should be bcc'd on this email? Multiple email addresses can be entered either individually, or as a comma... [more](#)

From

imahashi20221112@gmail.com ▼

Select an email address or alias from your Gmail Account. Defaults to the primary email address.

From Name

Imahashi's App

Reply To

Enter text or insert data...

Specify a single reply address other than your own.

Subject (required)

Welcome to my app

Body Type

plain ▼

If using the HTML option, you must add any and all formatting (paragraphs, lists, etc) directly using HTML. [Learn more](#).

Body (required)

Hello,  1. Full Name: Ryo Imahashi.

Thank you for signing up my app.

I hope you enjoy it.

- If the "to:" address is your real email address, click "Test action" to confirm receipt of the email.
  - If the address is not your real email address, do not click "Test action" (because the email will fail to be sent).
- Click on "Publish Zap"

The image shows two side-by-side screenshots of the Zapier 'Test action' step for a 'Send Email to Gmail' zap.

**Screenshot 1 (Left):** This screenshot shows the initial configuration of the test action. It includes a preview of the email body, which reads:

```
body: Hello, Ryo Imahashi.  
Thank you for signing up my app.  
I hope you enjoy it.  
  
body_type: plain  
from: imahashi20221112@gmail.com  
from_name: Imahashi's App  
subject: Welcome to my app  
to:  
1: imahashi@example.com
```

Below the preview, there is a section titled 'Empty fields:' with the following items:

- Cc: empty (optional)
- Bcc: empty (optional)
- Reply To: empty (optional)
- Signature: empty (optional)
- Label/Mailbox: empty (optional)
- Attachments: empty (optional)

**Screenshot 2 (Right):** This screenshot shows the result of the test action. It displays a success message and the details of the sent email:

**Send Send Email to Gmail**  
We'll use this as a sample for setting up the rest of your Zap.

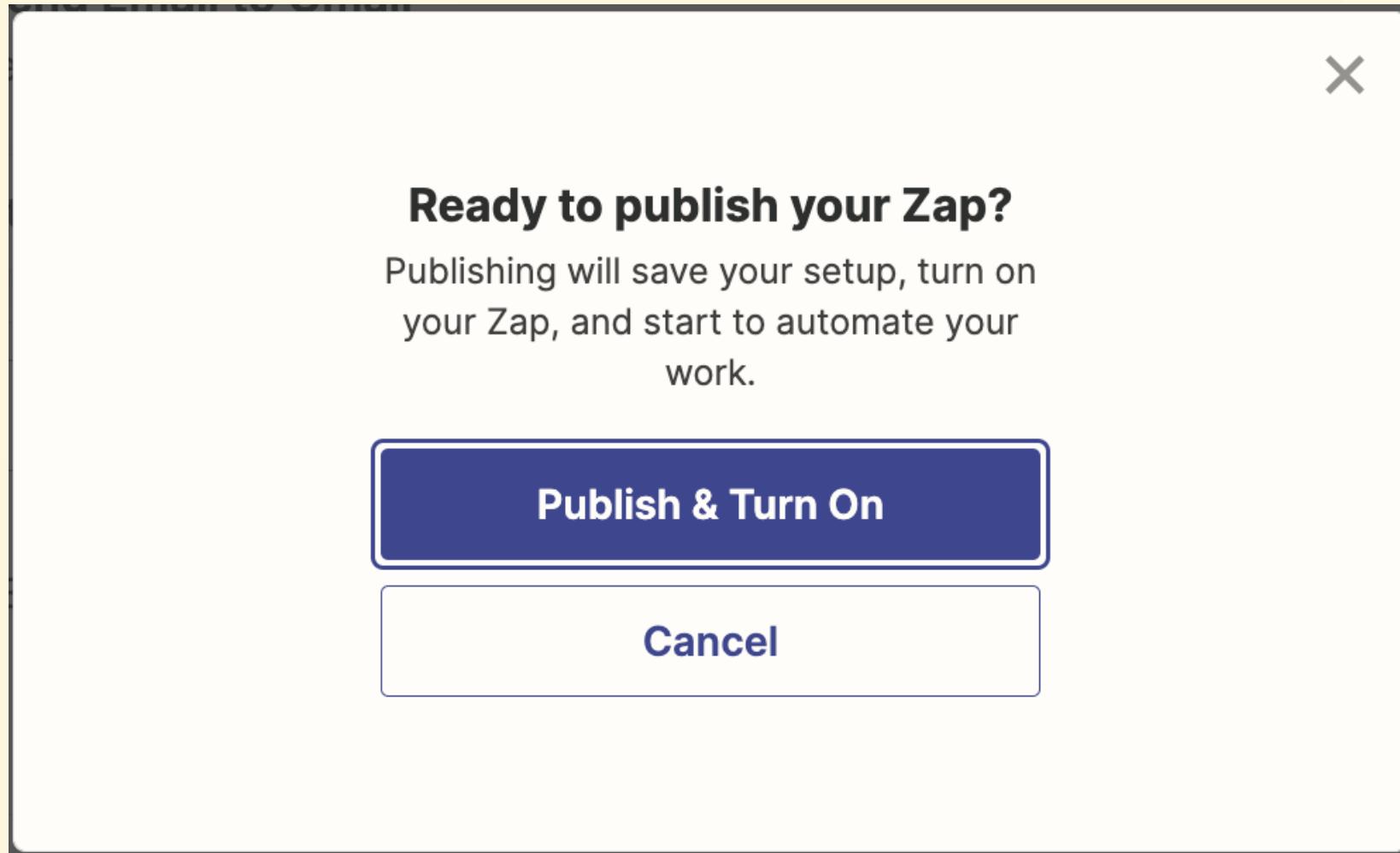
A message indicates: A Send Email was sent to Gmail about 2 minutes ago.

Below this, the email details are listed:

- id: 1846c01f2d6ddf3b
- threadId: 1846c01f2d6ddf3b
- labelIds:  
1: SENT

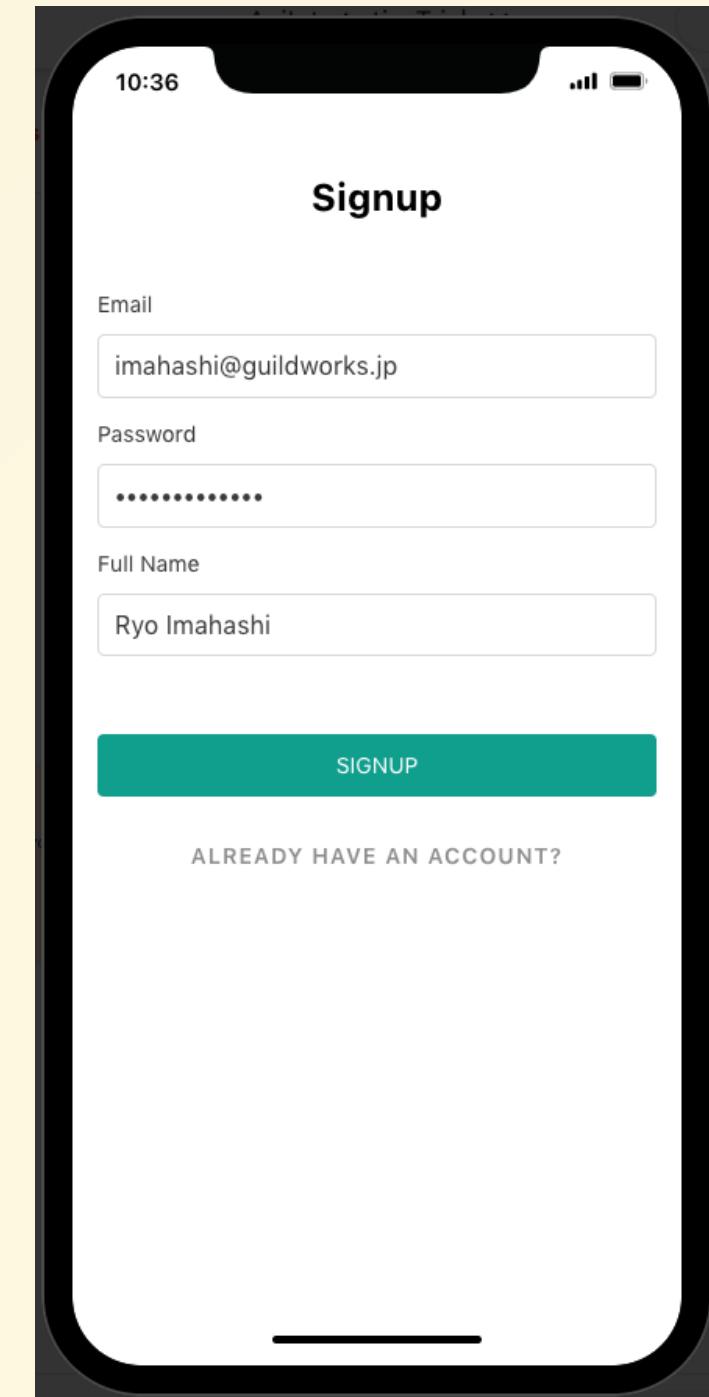
At the bottom, there are two buttons: 'Retest action' and a large blue 'Publish Zap' button.

- Click "Publish & Turn On"



Let's confirm that Welcome email will be sent in ApilntegrationTrial app.

- In the Preview function, enter your real email address and Signup



You will not receive an email immediately after Signup.  
For 14 days, Zap will run every 2 minutes for the free trial period of Zapier's Professional Plan. After that, you will be on the Free Plan and Zap will run every 15 minutes. (Reference: [Pricing](#))

Free	Starter	Professional
For individuals who need the basics of automation.	For individuals with simple, multi-step automation needs.	For individuals that need the full power of Zapier's automation platform.
\$0 USD free forever	\$19.99 USD /month, billed annually	\$49 USD /month, billed annually
100 tasks /mo	750 tasks /mo	2K tasks /mo
<a href="#">Current plan</a>	<a href="#">Upgrade</a>	<a href="#">Upgrade</a>

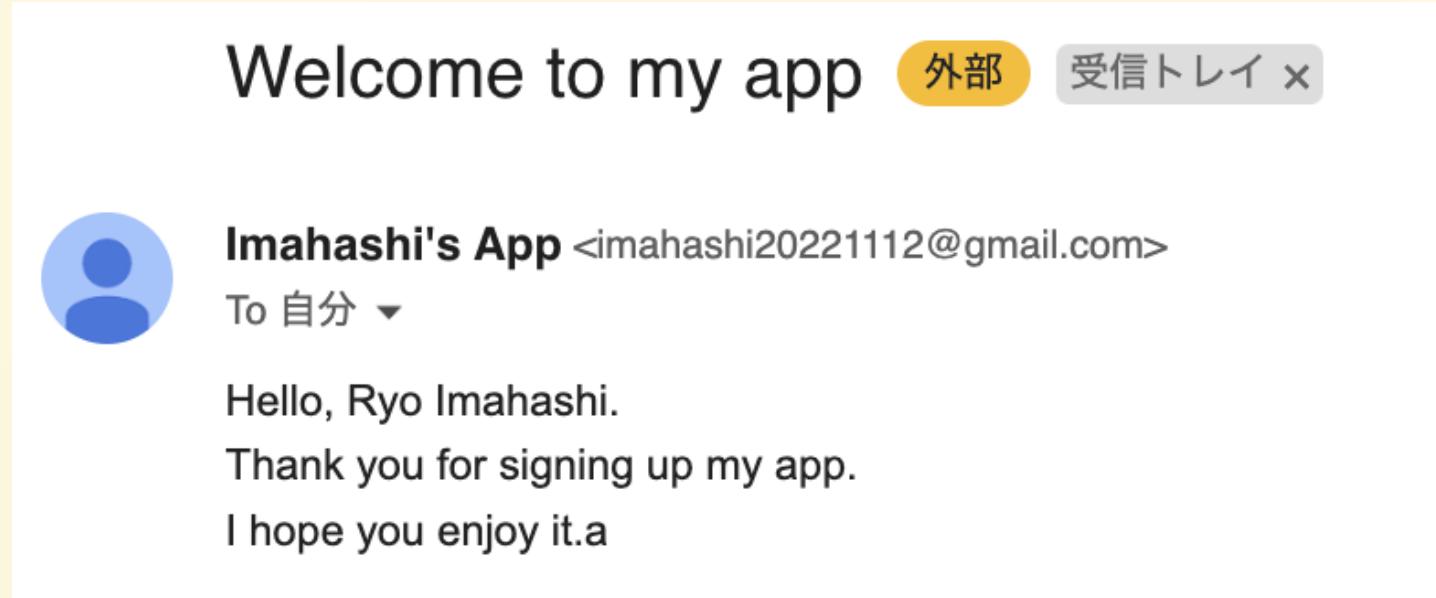
Free plan features	Starter plan features	How often Zapier checks for new data to start your Zap.
⌚ 5 Zaps ⌚ 15 min update time	⌚ 20 Zaps ⌚ 15 min update time	⌚ 2 min update time

You can also manually run a Zap immediately from the Zap listing screen.

- Go to <https://zapier.com/app/zaps>
- Select the Zap you want to run and click "Run"

The screenshot shows the Zapier web interface. On the left is a sidebar with icons for Dashboard, Zaps (which is selected and highlighted in orange), Transfers, My Apps, Zap History, Explore, Get Help, and Free Plan. The main area is titled 'Zaps' and 'Home'. It features a search bar, a 'Private folders' dropdown, and a table with columns: Name, Owner, Warnings, Last edit, and Running. A single Zap is listed: 'Name your Zap' by '20221112 I.', created 18m ago, with 'On' status. To the right of this Zap is a context menu with options: View, Details, History, Run (which is highlighted in blue), Rename, Share, Change ownership, and Duplicate.

Two minutes after Signup or after manually running the Zap, you will see that you have received a Welcome email at the email address you used for signing up.



The screenshot shows an incoming email with the following details:

- Subject:** Welcome to my app
- From:** Imahashi's App <imahashi20221112@gmail.com>
- To:** 自分 ▾
- Buttons:** 外部 (External) and 受信トレイ × (Inbox)
- Email Content:**

Hello, Ryo Imahashi.  
Thank you for signing up my app.  
I hope you enjoy it.a

# With Zapier, you can integrate many other services.

When you find something you can't achieve with Adalo alone, it may be good to think about whether you can achieve it by integrating your app with other services.

Reference: [Popular ways to use Adalo workflows](#)

## Popular ways to use Adalo workflows

Search for apps to connect to Adalo



Send Gmail emails for new Adalo records

by Zapier • Adalo + Gmail

Try it

...



Create Google Sheet spreadsheet rows for new Adalo records

by Zapier • Adalo + Google Sheets

Try it

...



Add new Adalo records to Airtable

by Zapier • Adalo + Airtable

Try it

...



Send Twilio texts for new Adalo records

by Zapier • Adalo + Twilio

Try it

...



Send Slack channel messages for new Adalo records

by Zapier • Adalo + Slack

Try it

...



Add Mailchimp subscribers from new records in Adalo

by Zapier • Adalo + Mailchimp

Try it

...



Add Webflow CMS items to new Adalo records

by Zapier • Adalo + Webflow

Try it

...



Add Adalo records for new Google Sheets spreadsheet rows

by Zapier • Google Sheets + Adalo

Try it

...

Load more

That's it for external integration of Adalo.

# Team Development with Adalo

Before exercises, let me tell you how to collaborate in Adalo's app development for team development in Development Phase.

Adalo has the ability to invite team members as co-editors, but this is only available with a paid plan.

The screenshot shows the Adalo mobile application interface. On the left is a vertical toolbar with icons for creating new items, adding filters, displaying data, saving, undoing, redoing, and viewing details. The main area is the 'Settings' screen, which includes sections for 'App Settings' and 'Display Settings'. A collapsed section for 'App Access' is shown, with a expanded section below it titled 'App Access' containing a list of users and an 'Add Team Member' button. A modal window is overlaid on the screen, featuring a large title 'Upgrade to Add Editors!', a descriptive message 'Switch to a paid plan to start collaborating on your app design.', and two buttons: 'BACK' and a teal-colored 'UPGRADE' button.

Settings

App Settings

Display Settings

App Access

Ryo Imahashi  
imahashi+20221105@guildworks.jp

Add Team Member

Invite New Team Members...

Upgrade to Add Editors!

Switch to a paid plan to start collaborating on your app design.

BACK

UPGRADE

Therefore, everyone in same team should log in with the same Adalo account to edit same app.

- When you start developing your team's apps on Adalo, create a Google account for your team and sign up for Adalo with that account

[https://accounts.google.com/signup/v2/webcreateaccount?  
continue=https%3A%2F%2Faccounts.google.com%2FManageAcco  
unt%3Fnc%3D1&dsh=S50453738%3A1637917137418951&biz=f  
alse&flowName=GlifWebSignIn&flowEntry=SignUp](https://accounts.google.com/signup/v2/webcreateaccount?continue=https%3A%2F%2Faccounts.google.com%2FManageAccount%3Fnc%3D1&dsh=S50453738%3A1637917137418951&biz=false&flowName=GlifWebSignIn&flowEntry=SignUp)

- Everyone should log in using that email address and password for collaborative editing

# Notes on collaborative editing in Adalo

- Edits made by others will not be reflected on your screen in real time. You need to reload to reflect them.
- If you edit the same screen at the same time, the edits made earlier will be overwritten by the edits made later, and the screen will be in the state of the last person who made the edits.
  - The earlier edits will be undone, even if the edited components are different.
- For Actions, if you add them to the same screen at the same time, only the later one will be saved.
- Database, too, will be overwritten with the state of the last person who edited it if it is edited at the same time.

- If you edit on different screens at the same time, both will be saved and you will be fine. When editing, share with your team members which screen you are editing. It is also recommended to decide who is in charge of each screen.
- Before switching the screen to be edited, reload the screen to reflect the latest state (this will prevent you from reverting to the old state when someone else has edited the screen).
- To avoid inadvertently touching other people's screens, it is also recommended that you develop the screens apart from each other on Canvas.
- For the Database, decide on one person to be in charge and let that person update the database.

This is it about how to do team development with Adalo.

When you start team development in Development Phase, please check the above notes while actually doing collaborative editing with team members. (Because it is sad to see your work lost 😢 )

# Exercises

1. Develop a team member management app with following features.
  - Registration of team members
  - Display of member list
  - Display member details
  - Update member information
  - Delete a member
  - Your original functions
2. Develop new app freely (if there is enough time after exercise 1).

At the end of the excersises, each one of you will present the app you made in the exercise.

When the app is ready to use, share the URL on Slack for everyone to see.

## Reference

- There are a number of Adalo apps available that can be cloned, so you may want to look for one that is similar to what you want to do.
  - App Templates  
<https://www.adalo.com/app-templates>
  - UI & Functional Kits  
<https://www.adalo.com/cloneable-kits>

## Examples of apps that can be cloned

- Event Calendar <https://www.adalo.com/cloneables/event-calendar>
- SNS follow function <https://www.adalo.com/cloneables/follow-function>
- Facebook clone <https://www.adalo.com/cloneables/facebook-clone>
- Blog app <https://www.adalo.com/cloneables/minimal-blog-app>
- Ecommerce app <https://www.adalo.com/cloneables/ecommerce-app>



# Presentation of the exercise results

Please present the application that you made in the exercises.

# Summary

- In this lecture, you designed a database and made it possible to perform CRUD operations on the database of sample application.
- You learned some of Adalo's features and improved sample app.
- You also learned how to integrate your adalo app with external services with following methods.
  - Marketplace external integration component
  - Custom Action
  - External Collection
  - Connector service

- Based on what we've learned so far, I think it's a good idea for teams to think about whether you can create your application using Adalo or not.
- Next time, We will give a lecture about Bubble, a no-code tool.  
Look forward to it!

**That's all!**

**Thank you for your hard work!**