

[Blog](#)[About Bubble](#)[The No-Code Manif](#)[Get started](#)[HOW TO](#)

# How To Build A Duolingo Clone With No Code

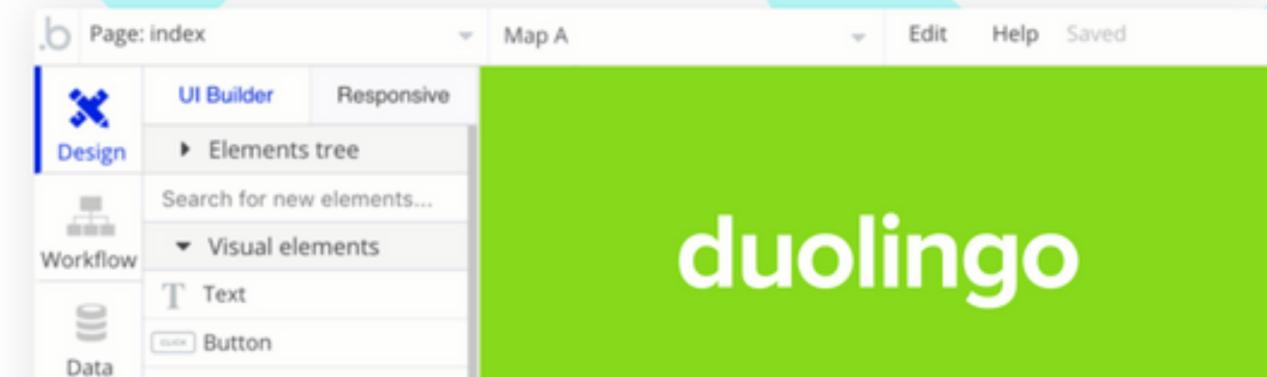
Learn how to create a language learning app without writing code, using Bubble's visual web editor.



LACHLAN KIRKWOOD

16 JUL 2020 • 11 MIN READ

## How to build Duolingo on Bubble





If your answer is ‘oui’ (yes), then you’re in the right place!

It’s never been easier to start building powerful software with [Bubble](#). By using our visual programming tool, it’s possible to quickly design and build working software that goes beyond static experiences. Makers from around the world are using Bubble to build marketplaces, directories, and even social networks.

Throughout this post, we’ll uncover the process behind building a Duolingo clone using Bubble. Whether you’re looking to replicate the entire product, or you’re just interested in utilizing some of the core features from the platform, this guide will share how to start writing the workflows for your product.

The steps to building a Duolingo clone with no code include:

#### [Get Started](#)

[Register your account on Bubble](#)

[Configuring your database](#)

[Building workflows](#)

[Creating a new lesson word](#)

[Sending data between pages](#)

[Displaying a language lesson](#)

[Marking an answer as correct or wrong](#)

[Launch](#)

[Templates](#)

[Start Building](#)

## Get Started

[Blog](#)[About Bubble](#)[The No-Code Manifesto](#)[Get started](#)

## Register your account on Bubble

We also recommend completing our [introductory lessons](#), which will guide you through the step-by-step process of the most common Bubble features. This will help you get a running start when building your Duolingo clone.

When kickstarting your project, you can choose to begin by wireframing your product design, or building the necessary fields within your database.

In this case, we'll start by using Bubble's visual design tool to shape the user-interface of our platform. If you're replicating a version of Duolingo, some of the core pages you'll want to include are:

- A backend portal to **create new lessons**: The process of adding a new lesson will include unique pages to select a language, a category, and also add new words
- A **language selection page**: Where users can select which language they'd like to learn
- A **category selection page**: Allowing users to select from a list of lesson categories within a language
- A **lesson page**: Displaying a core lesson

The image displays two wireframe prototypes side-by-side. The left wireframe is titled "Choose a language" and shows three options: French (represented by a blue square), Italian (represented by a green square), and Spanish (represented by a red and yellow horizontal bar). The right wireframe is titled "Which one of these is a Dog" and presents four items for identification: "chatte" (orange cat), "chienne" (brown dog), "poisson" (yellow fish), and "oiseau" (blue bird). Each item is accompanied by a small cartoon illustration.

[Blog](#)[About Bubble](#)[The No-Code Manifesto](#)[Get started](#)

A major feature within Bubble is the ability to send data between pages. This allows you to create one generic version of a page, then dynamically display the relevant content from your database when it's required.

In the case of your Duolingo clone, you'll only need to create one page to host each individual lesson. We can then write the necessary logic to display only the relevant content on each page when it's needed (we'll cover this in more detail soon).

## Configuring your database

Once you've mapped out the display of your product, you can focus on creating the necessary data fields to power your application. We'll rely on these fields to connect the workflows behind your product.

Bubble's pre-built database makes it easy to create different data types with unique fields. When building Duolingo as an MVP, we'll need to create the following data types & fields:

### Data type: User

#### Fields:

- Name
- Languages learning - List of languages. **Note:** *Creating a field as a list based on a separate data type allows you to seamlessly integrate all of its relevant data fields.*

The screenshot shows the Bubble database configuration interface. At the top, there is a 'Type name' input field containing 'User'. Below it, there is a 'Languages learning' field set to 'List of Languages'. Under the 'User' type, several fields are listed: 'Name' (text, default), 'email' (text, Built-in field), 'Modified Date' (date, Built-in field), and 'Created Date' (date, Built-in field). Each field has a small trash bin icon to its right.

[Blog](#)[About Bubble](#)[The No-Code Mani](#)[Get started](#)

## Data type: Language

### Fields:

- Name
- National flag
- Categories - List of categories

The screenshot shows the 'Language' data type configuration in the Bubble interface. At the top, there is a 'Type name' input field containing 'Language'. Below it, there is a 'List of Categorys' section with a 'Categories' input field and a 'List of Categorys' output field. There are also three rows for fields: 'Name' (text), 'National flag' (image), and 'Creator' (User). Each row has a 'default' input field and a 'Upload' button. At the bottom, there is a 'Create a new field' button.

## Data type: Category

### Fields:

- Category title
- Language
- Image
- Words - List of words

The screenshot shows the 'Category' data type configuration in the Bubble interface. At the top, there is a 'Type name' input field containing 'Category'. Below it, there is a 'Category title' input field and a 'text' output field. There is also a 'default' input field and a 'Delete' button at the bottom right.

**Words** List of Words

Creator	User	Built-in field
Modified Date	date	Built-in field
Created Date	date	Built-in field

Create a new field

## Data type: Word

### Fields:

- English spelling
- Foreign spelling
- Is correct - With default as no
- Category - Category
- Picture

Type name **Word**

Category	Category	default	
English spelling	text	default	
Foreign spelling	text	default	
Is correct	yes / no	default	no
Word picture	image	default	Upload
Creator	User	Built-in field	
Modified Date	date	Built-in field	
Created Date	date	Built-in field	

Create a new field

## Building workflows

Now that you've structured both the design and database of your application, it's time to

[Blog](#)[About Bubble](#)[The No-Code Mani](#)[Get started](#)

In Bubble, the main way to do this is with workflows. Each workflow happens when an event occurs (e.g. a user clicks on a button), and then runs a series of “actions” in response (e.g. “sign the user up”, “make a change to the database”, etc.)

## Creating a new lesson word

One of the first features we'll build is a back-end portal where admins can upload new words as lessons to our platform.

**Note:** Before this point, I'd recommend manually adding a list of languages and categories to your platform through your bubble database. These categories could include topics like basics, travel, or people, and will be stored within each language.

Create a new database entry

Type of thing: Language

Categories:  Add

Name:

National flag:

Upload

CREATE

Once our platform has some basic data to work off, we'll start by adding a page to select which language we'd like to create a lesson for.

The **upload-language** page will be used to first select a language we'd like to contribute to.

On this page, we'll add a repeating group element to display a list of all the languages within our database.

The data type of this repeating group will need to be configured as a **language**, and the data source will search for a list of all the languages available.

Choose a language

Current cell's Language's National flag

Search for Languages

Type: Language

RepeatingGroup Language

Appearance Conditional Transitions

Type of content: Language



Blog

About Bubble

The No-Code Mani

Get started

Now you're ready to start structuring the dynamic content that will be displayed within each grid. Simply map out the top row with the relevant content you'd like to show, then this powerful element will populate the remaining columns based on your existing app data.

## Sending data between pages

Within a repeating group, it's also possible to create events based on elements in each individual row. This feature will become useful when building navigational features across your platform.

Once an admin has selected a language, we'll need to display a list of the relevant categories within a language that they can contribute to.

To build this workflow, we'll create a new trigger when the languages title is clicked.


[Blog](#)
[About Bubble](#)
[The No-Code Manifesto](#)
[Get started](#)

National flag

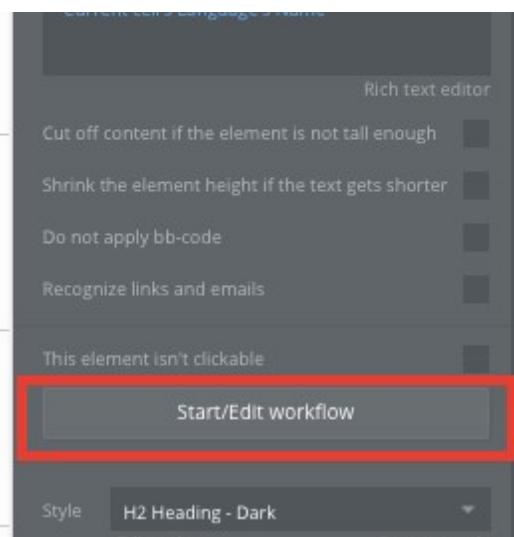
Current cell's Language's...

Current cell's  
Language's  
National flag

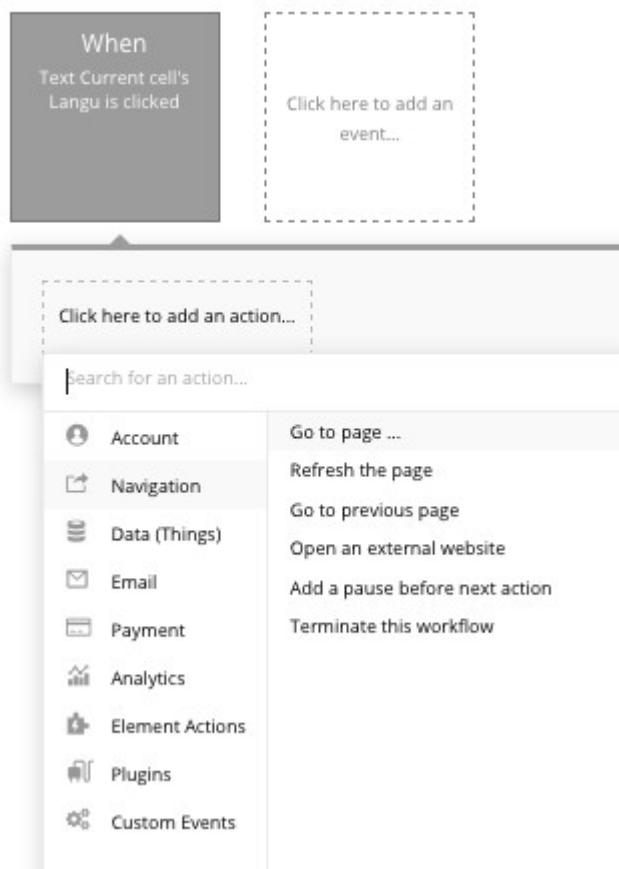
Current cell's Language's...

Current cell's  
Language's  
National flag

Current cell's Language's...



When creating this workflow, use a navigation event to send a user to another page.



From here, select the destination page type to be the **upload-category** page.

You'll then need to send additional data to this page so the Bubble editor knows which categories to display. The data you'll need to send is that of the **current cell's language**.

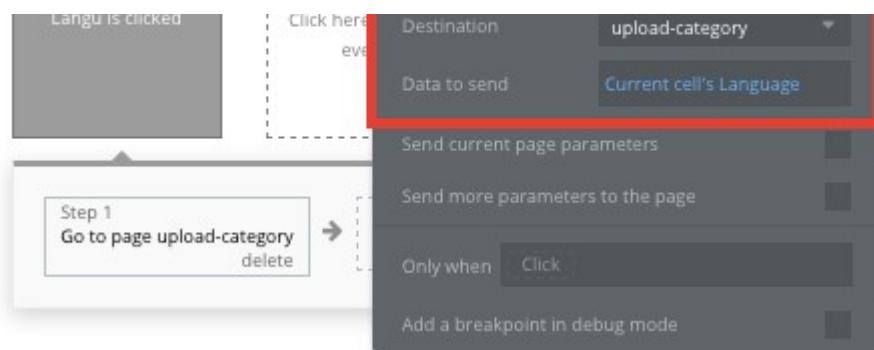


Blog

About Bubble

The No-Code Mani

Get started



When you're directed to the category page, you can easily pull this event data and display the content that's relevant to this language.

To build this function, you'll first need to ensure that the destination page type matches the data property that you're sending within the workflow. In this case, you'll need to set the **upload-category** to a **language** property.

The screenshot shows a "Choose a category" page with three items, each labeled "Current cell's Category...". To the right, the "upload-category" page settings are shown, with the "Type of content" set to "Language".

Next, we'll need to configure the repeating group on this page to display the list of categories where the language is equal to the current pages language.

The screenshot shows the "Search for Categories" configuration and its data source "RepeatingGroup Category". The "Search for Categories" section has a condition "Language = Current Page Language". The "RepeatingGroup Category" data source uses "Search for Categories" as its data source.



Blog

About Bubble

The No-Code Mani

Get started

Once a user has selected a category to contribute to, we'll build another navigation workflow, sending them to a portal where they can upload a new lesson (our upload-word page).

On our **upload-word** page, we'll need to configure the page type to be a category, then add a combination of input elements; an image uploader, and a checkbox.



Blog

About Bubble

The No-Code Mani

Get started

Once we've added the content of a new word into the on-page input, we'll trigger a new workflow when the **create button is clicked**.

Within this workflow, we'll start by **creating a new thing** - a new word.

The screenshot shows the Bubble workflow editor interface. At the top left is a grey box labeled "When Button Create is clicked". To its right is a dashed box labeled "Click here to add an event...". Below these is a large dashed box labeled "Click here to add an action...". Underneath this box is a search bar with the placeholder "Search for an action...". A sidebar on the left lists categories: Account, Navigation, Data (Things), Email, Payment, Analytics, Element Actions, Plugins, and Custom Events. The "Data (Things)" category is currently selected. To its right, under "Create a new thing...", there are several options: "Create a new thing...", "Make changes to thing...", "Make changes to a list of things...", "Delete thing...", "Delete a list of things...", "Copy a list of things...", "Download data as CSV", "Upload data as CSV", "Delete an uploaded file", and "Install more data (things) actions...".

We'll then need to start adding the relevant fields within our database. Map each of the on-page input elements with their corresponding field in your database.

The screenshot shows the "Create a new Word..." dialog box. On the left is a grey box labeled "When Button Create is clicked". Below it is a white box labeled "Step 1 Create a new Word... delete". To the right of the dialog box is a red-bordered area containing four field mappings:

- English spelling = Input - English word's value
- Foreign spelling = Input - Foreign word's value
- Word picture = PictureUploader A's value
- Category = Current Page Category

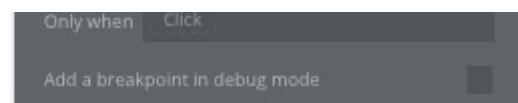


Blog

About Bubble

The No-Code Mani

Get started

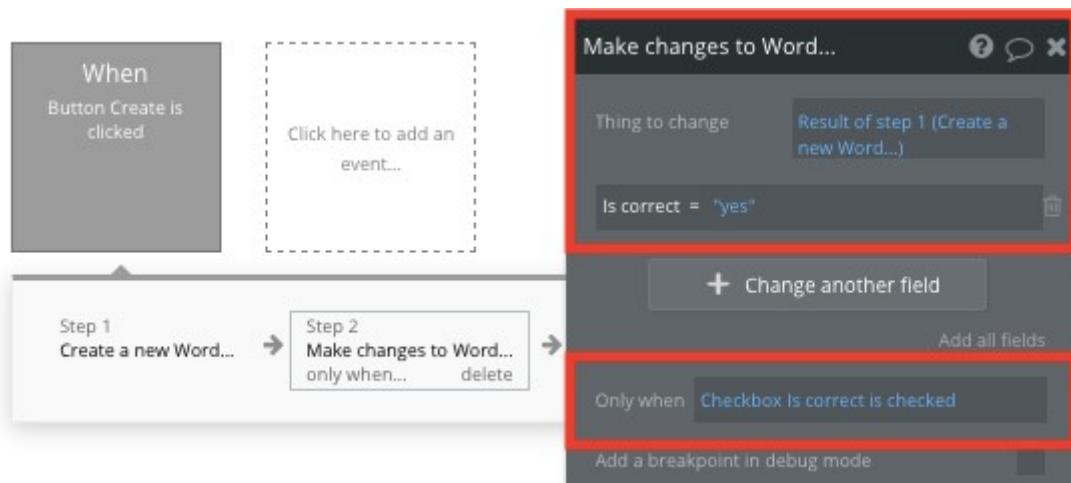


When creating a new lesson word, we'll need to start by adding the correct version of a word first. As the word is correct, we'll make sure to select our '**is correct**' checkbox element.

To register this as the correct word in our lesson, we'll need to **make changes** to the word within our workflow - updating its 'is correct' status.

Note: When adding this event in your workflow, it's important to add a conditional trigger, allowing it to only run if our checkbox has been ticked.

If the word has been marked as correct, we'll then update its status to '**yes**'.



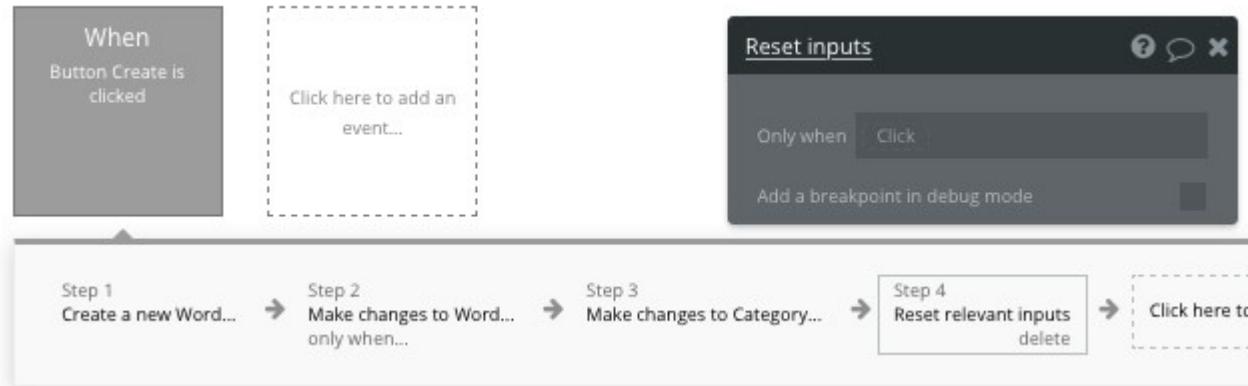
Once a word has been created in our workflow, we'll then need to add this term to our relevant category.

Once again, we'll opt to **make changes to a thing** - the **current pages category**, adding to it the new word we just created.



[Blog](#)[About Bubble](#)[The No-Code Mani](#)[Get started](#)

The final step we'll add to this workflow is a function to reset all of the current page elements, allowing you to easily add an alternative word that isn't correct into this lesson.



## Displaying a language lesson

Once we've created a lesson on our back-end portal, we can focus on building the front-end experience for our end users.

The process of building the user workflows will follow a similar approach to our previous steps.

We'll start by once again creating a dedicated screen for users to select from a list of languages they'd like to study.

To streamline the process of building this page, you can even duplicate our existing [upload-language](#) page.

### Languages

Current cell's Language's National flag  
Current cell's Lan...

Search for Languages

RepeatingGroup Language

Appearance Conditional Transition



Blog

About Bubble

The No-Code Mani

Get started

On this page, we'll once again create a new workflow when a languages flag is clicked.

This workflow will send a user to our **category-select** page, sending with it the **current cell's language**.

We'll also want to include an additional event, adding the language itself to a users list of languages they're learning.

For this step, we'll need to **make changes to a thing** - the **current user**. The thing we'll want to change is their list of **languages learning**, adding to it the **current cells language**.



Blog

About Bubble

The No-Code Mani

Get started

On our category selection page, we'll configure the page type to be a **language**, allowing us to reference its unique categories we've created.

The screenshot shows a repeating group of categories on the left and a page settings panel on the right.

**Repeating Group (Left):**

- Header: Categories
- Content: Current cell's Category's Image, Current cell's Category's
- Content: Current cell's Category's Image, Current cell's Category's
- Content: Current cell's Category's Image, Current cell's Category's
- Content: Current cell's Category's Image, Current cell's Category's

**Page Settings Panel (Right):**

- Header: category-select
- Appearance tab selected.
- Page title: Your Bubble app
- This page is a native app
- Mobile version
- Type of content: Language (highlighted with a red box)
- Field for readable URL
- Style
- Background style: Flat color
- Background color: #FCFCFC
- Width: 562, Height: 653

Within our repeating group, we'll search for a list of available categories where the language equals the **current page's language**.

The screenshot shows a search dialog and a repeating group configuration side-by-side.

**Search Dialog (Left):**

- Type: Category
- Language = Current Page Language (highlighted with a red box)
- Add a new constraint
- Sort by
- Ignore empty constraints
- Close, Delete

**Repeating Group Configuration (Right):**

- Header: RepeatingGroup Category
- Appearance tab selected.
- Type of content: Category
- Data source: Search for Categories (highlighted with a red box)
- Layout style: Vertical scrolling
- Rows: 4, Columns: 1
- This repeating group has a fixed width
- Cell min width (current width: 468px): 100

Now, when a category is selected, we'll create one last navigation event, sending the user to our **lesson-word** page.

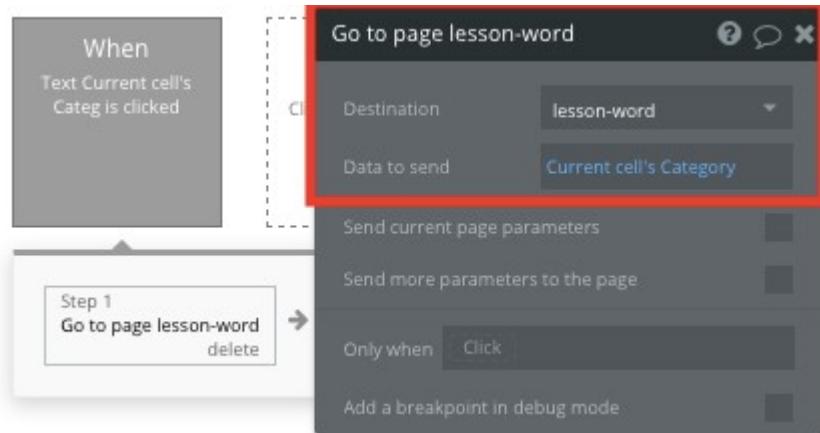


Blog

About Bubble

The No-Code Mani

Get started



Now, on our **lesson-word** page, we'll start by configuring the page type to be a category.

The screenshot shows the configuration for the 'lesson-word' page. Under the 'Appearance' tab, the 'Page title' is set to 'Your Bubble app'. In the 'Type of content' dropdown, 'Category' is selected and highlighted with a red box. Other options like 'List' and 'Table' are visible in the dropdown menu.

Next, we'll start adding dynamic elements to our page to display the current lessons featured word.

As taken added our lessons featured word first through our upload page, we'll opt to display the first item from the **current pages category**.

The screenshot shows a 'Text' block with the placeholder text 'Which one of these is a Current Page'. This block is part of a larger card-based interface where other fields like 'Current cell's Word's Word picture' are also present.

The screenshot shows a 'Text' block with the placeholder text 'Text Which one of these i...'. The text content of the block is: 'Which one of these is a Current Page Category's Words:first item's English spelling'. This block is also part of a card-based interface.



Blog

About Bubble

The No-Code Mani

Get started

Next, we'll add another repeating group to the page, only this time, we'll set the content type to be a **word**.

When configuring the data source of this repeating group, we'll search for a list of words where the category equals the same value as the **current pages category**.

As the content within the repeating group will display a list of potential answers to our lesson, we'll also opt to shuffle each of the items within our repeating group in a random order.

Within our repeating group cell, we'll then add a word's foreign spelling, and an image to associate with the word itself.



Blog

About Bubble

The No-Code Mania!

Get started

## Marking an answer as correct or wrong

Once a user takes part in a lesson, we'll need to display if they've selected the right answer to the current question.

Using **icon elements**, we can easily start to display both a correct and wrong label that will be triggered upon a specific answer.

As we'll only need to display these icons when an answer has been selected, we'll ensure these icons aren't visible when the page is initially loaded.

Next, we'll need to create a new workflow when the text of a repeating groups cell is clicked.



Blog

About Bubble

The No-Code Mani

Get started

**Current cell's Word**

Current cell's Word's Word picture

**Current**

Current ce picture

Rich text editor

Cut off content if the element is not tall enough

Shrink the element height if the text gets shorter

Do not apply bb-code

Recognize links and emails

This element isn't clickable

Start/Edit workflow

Style: H2 Heading - Dark

Edit style Remove style

Tooltip text (on hover)

This element is visible on page load

Make this element fixed-width

For this workflow, we'll set a condition to the trigger of the text. In this instance, our condition will define if the **current cells word** is indicated with a **correct = 'yes'** data field.

When

Text Current cell's Word is clicked and Current cell's Word's Is correct is "yes"

Click here to add an action

Text Current cell's Word is clicked

Element: Text Current cell's Word

Event color: Gray

Workflow folder:

Only when Current cell's Word's Is correct is "yes"

Add a breakpoint in debug mode

Disable workflow

Once this trigger has been activated, we'll then choose to **show an element** - our green icon.

When

Text Current cell's Word is clicked and Current cell's Word's Is correct is "yes"

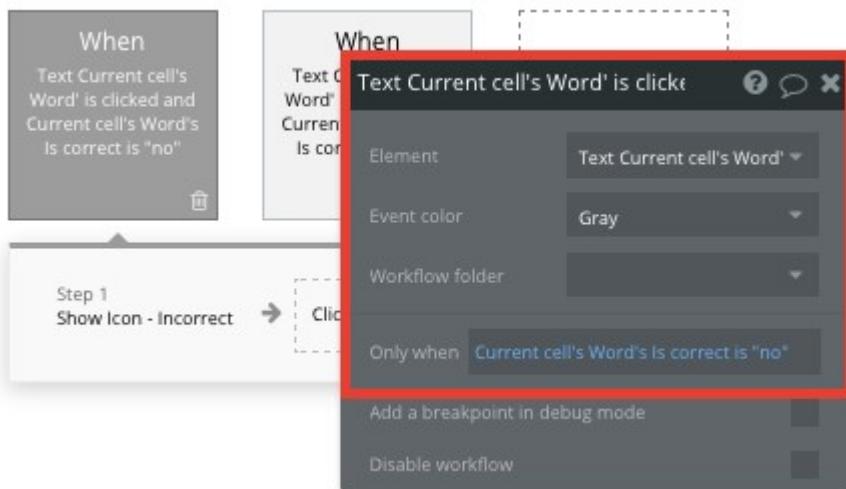
Click here to add an event

Show

Element: Icon - Correct

[Blog](#)[About Bubble](#)[The No-Code Mani](#)[Get started](#)

We'll then replicate this same workflow, only this time, we'll update the condition to identify when the **word is correct = 'no'**, resulting in displaying the **wrong icon**.



## Additional features

Now that you're familiar with building custom data fields and displaying dynamic content, you can start getting creative with the experiences you build within your product. Additionally you can:

- Build an experience bar as users complete new lessons
- Display a list of a user's lessons on a profile page
- Add sound files to each word, allowing users to hear the pronunciation for each lesson. **Note:** We've previously covered how to work with audio files in our [How To Build Spotify guide](#).

## Privacy & Security

Now that you have the basics of your app, don't forget to start setting some [privacy rules](#) and conditionals to keep your data secure - starting with roles in the '**Privacy**' section of your [Data tab](#). You can also check if you're unintentionally exposing any data with an [API checker](#).

[Blog](#)[About Bubble](#)[The No-Code Manif](#)[Get started](#)

thousands of dollars. As you launch and grow in users, our paid plans allow you to host the product on your own custom domain, starting from as low as \$25 per month. Some templates and plugins may cost more, but you can build all the functionality of your Duolingo clone without any additional costs.

## Templates

If you didn't want to build your Duolingo clone from scratch, you can purchase one of the templates made by our community members. Some similar templates include:

- [Courses like Udemy](#) by Zeroqode

## Start Building

Bubble can help you build a Duolingo clone or any other product you choose! It's never been easier to build something incredible without having to code.

Ready to join the no-code movement? Register your free account [here](#) and get started today.

If you'd like additional help when building your product, our thriving [community forum](#) is a great place to source actionable advice and resources. Any maker is also encouraged to ask questions within the community to help streamline their no-code journey.

**Get started with a free account.**

  
Your email address[Get started](#)

[Blog](#)[About Bubble](#)[The No-Code Mania!](#)[Get started](#)

Very cool! We actually just finished building a language learning site for Amharic using Bubble -

[www.fi-dal.com](http://www.fi-dal.com) 😊

**tetepierre69**

Sep '20

Certainly a great tutorial. But for brains like mine it would have been even easier if it had been a video tutorial 😢. After going through it i honestly didn't get everything and unfortunately could not build my duolingo clone.

**rizal.adanza**

10 Feb

Can you show the sample database data you are working off of. Something isn't working in the tutorial. When I click on a language it goes to the page of categories but the data shows my categories are empty.

## Continue Discussion

### How To Build A Nextdoor Clone With No Code

14 Oct 2021 – 10 min read

### How To Build A Headspace Clone With No Code

11 Oct 2021 – 11 min read

### How To Build A Zendesk Clone With No Code

12 Apr 2021 – 9 min read

[See all 52 posts →](#)

**.b Forum**

Building a GRID document is as easy as 1-2-3

1. Importing a spreadsheet from a cloud service
2. Build a user interface
3. Share with colleagues and the entire world

## ECOSYSTEM

### GRID Review: Enhanced Spreadsheet & Doc Tool

Evaluating GRID as a no-code spreadsheet tool, based on features, functionality, pricing, alternatives and comparison.

**BUBBLE**

17 JUL 2020 • 4 MIN READ

**bubble**



[Blog](#) [About Bubble](#) [The No-Code Manif](#)

[Get started](#)

## No-Code Deep Learning with Bubble & Petarion

Build an app using an AI model - all without writing a single line of code.



ALLEN YANG

16 JUL 2020 • 3 MIN READ

© 2020, Bubble Group, Inc. All Rights Reserved.

[Latest Posts](#) [Bubble Home](#)