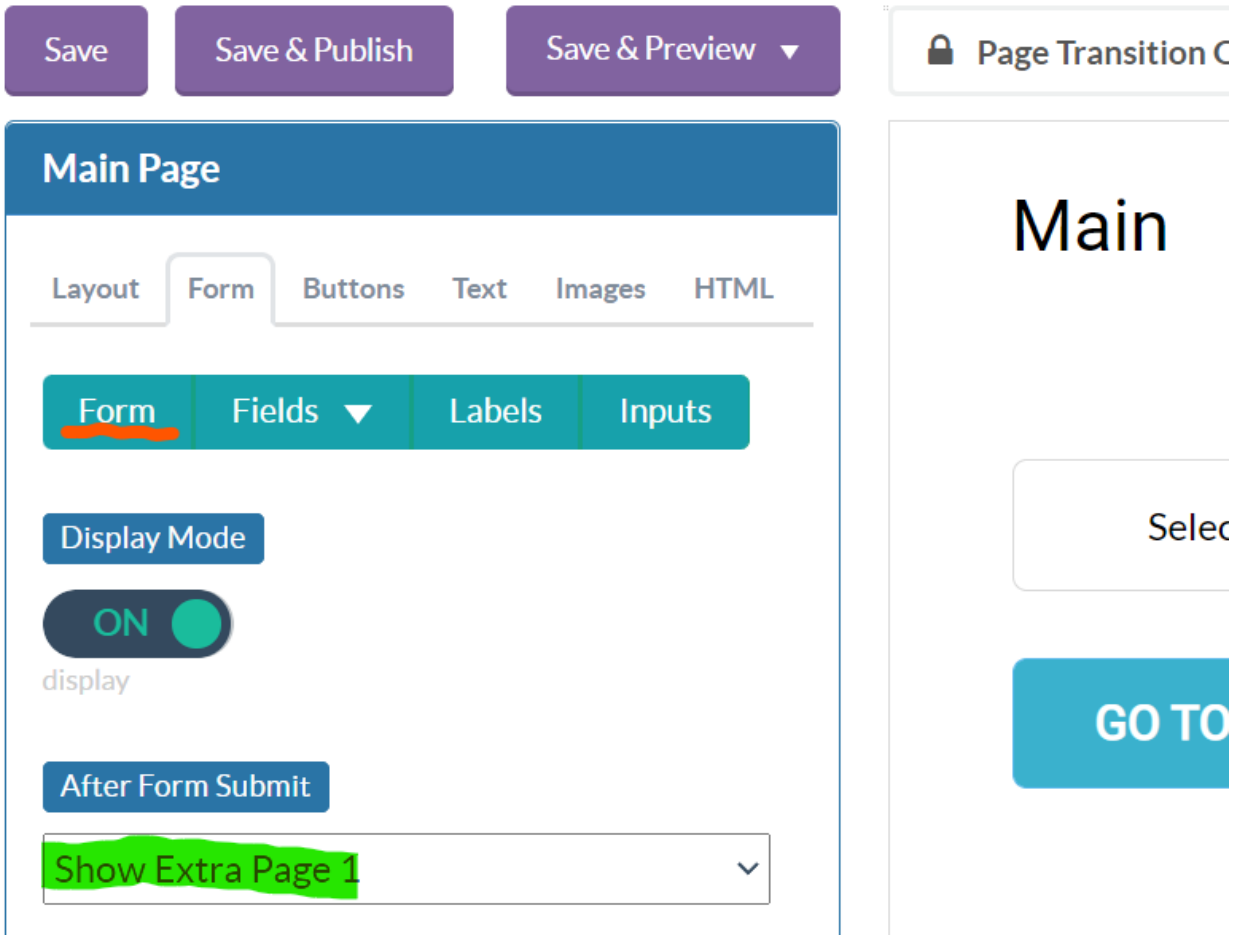
Configuring the Dynamic Page Change Override Extension

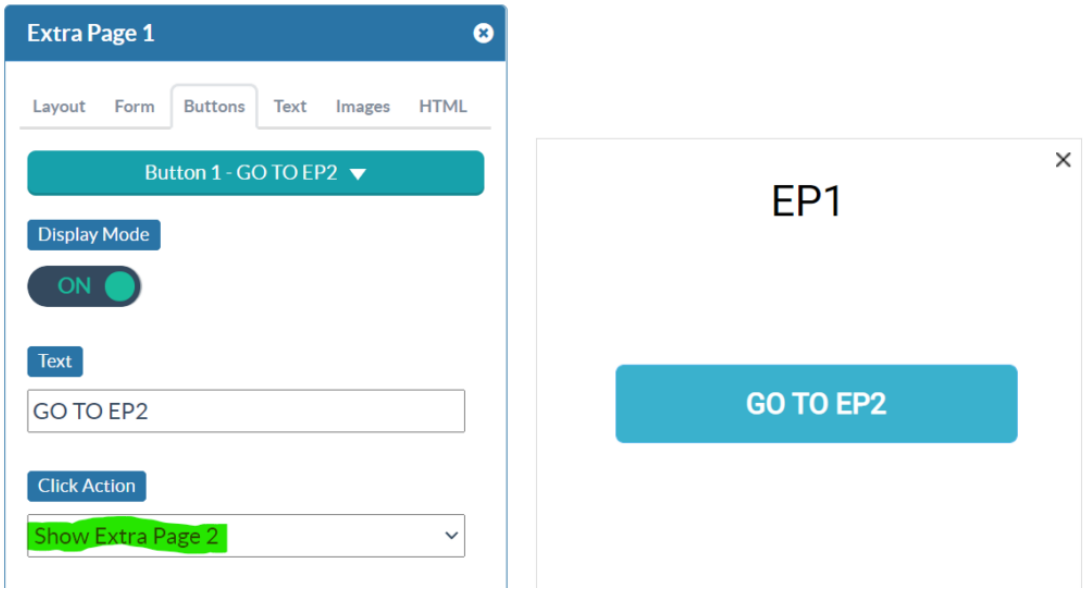
Last Modified on 06/24/2026 11:59 am EDT

Normally, page transitions in Digioh Campaigns are "statically" configured, based on the configured After Form Submit action or a Button Click action.

For example, a form on main would have a button with action "Submit Form" and the Form is configured to show another page:



Or if the page does not have a form, or you allow skipping of submission, Buttons can have a change page action:



Overriding Page Transitions

If you need to override the configured page transition, you can do so based on any prior submitted form data for that campaign. We have an app for that!

From the profile menu, top right, go to Custom JS (Apps) and install app #52 Dynamic Page Change Override, just once. This app is "always on", and will activate when you configure your Campaigns with specific metadata.

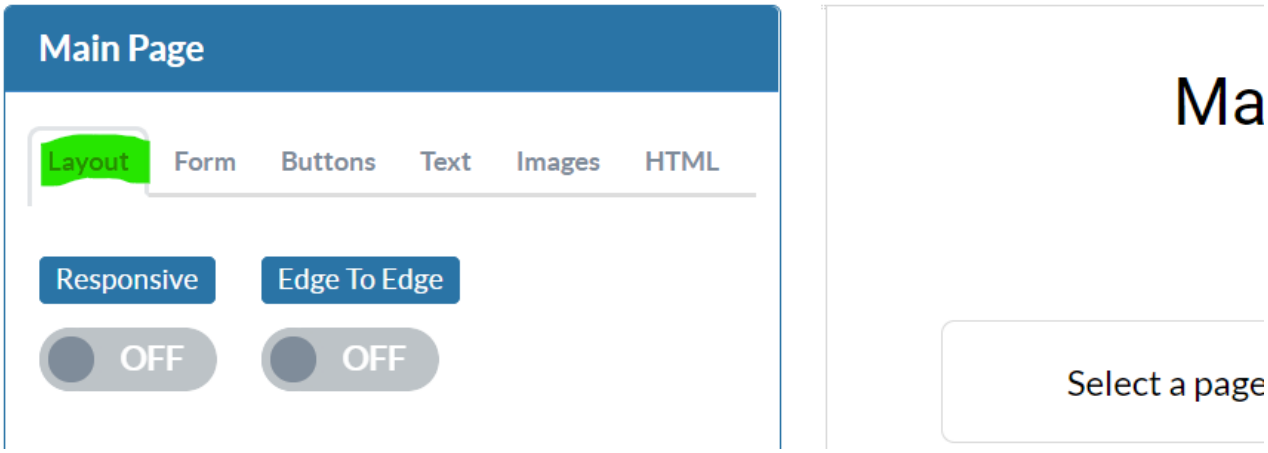
Normally, you configure a form or buttons on the *from* page, and it will transition *to* another page. The way to think about this app is that on the transition-*to* or -*from* page, you can explicitly configure metadata that will abort the transition and jump to another page of your choosing, based on prior form submission values.

Let's say we have a dropdown Field on the Main page, with submit values configured as follows:

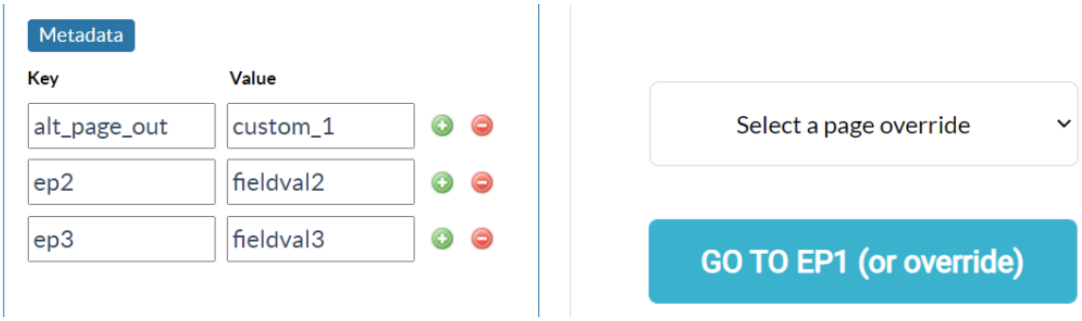


The form submit action will transition to Extra Page 1, and we want to potentially override this based on what the user selects in this dropdown, and we do this with metadata on the transition-*from* page which is Main.

Page-level metadata is configured at the bottom of the page's Layout tab:



Scroll down, and our page override metadata is as follows:



The "app activation" metadata command key is **"alt_page_out"**, and you set the value to be the *name* of the field you want to use dynamically to determine page routing. Field names can be:

- opt_in
- custom_1
- custom_2 ...
- and email, phone, name, first_name, last_name, full_name - but these are unlikely to be useful for branching

Typically you would use checkboxes, radios, or dropdown field types to control page transition overrides.

The app will grab the value of the named field and compare it to other metadata configured for this page. In the example above, we have metadata values "fieldval2" and "fieldval3", which corresponds to custom_1 dropdown submit values shown further above. The metadata keys are the names of the transition-to page that will override the Campaign config.

Valid page names are: main, thx, ep1, ep2, ep...

Innie or Outie?

The example above used the metadata command **alt_page_out** configured on the transition-from page, which implements a form of "branching" for transitions *out* of that page. You would typically use this early in your progressive form flow, where the user is making choices that affect their path through the Campaign experience.

If instead of "branching", your use case is closer to "skipping" a page, then you can use the alternate metadata command **alt_page_in**. This works exactly the same as alt_page_in, but it activates on transitions *in* to that page. For example, you might have a Main page with an opt_in checkbox to take a survey. On submitting, it transitions to Extra Page 1 which is the survey, and you have a alt_page_in=opt_in, with ep2=false. This will skip Extra Page 1 unless the user checked opt_in.

You can implement branching and skipping with either command, but alt_page_in can be more convenient and intuitive if, for example, you have multiple paths into a single page and you need to skip it in some situations.

Running into an issue or have a question? Reach out to our support team via support@digioh.com and we'll be happy to help!