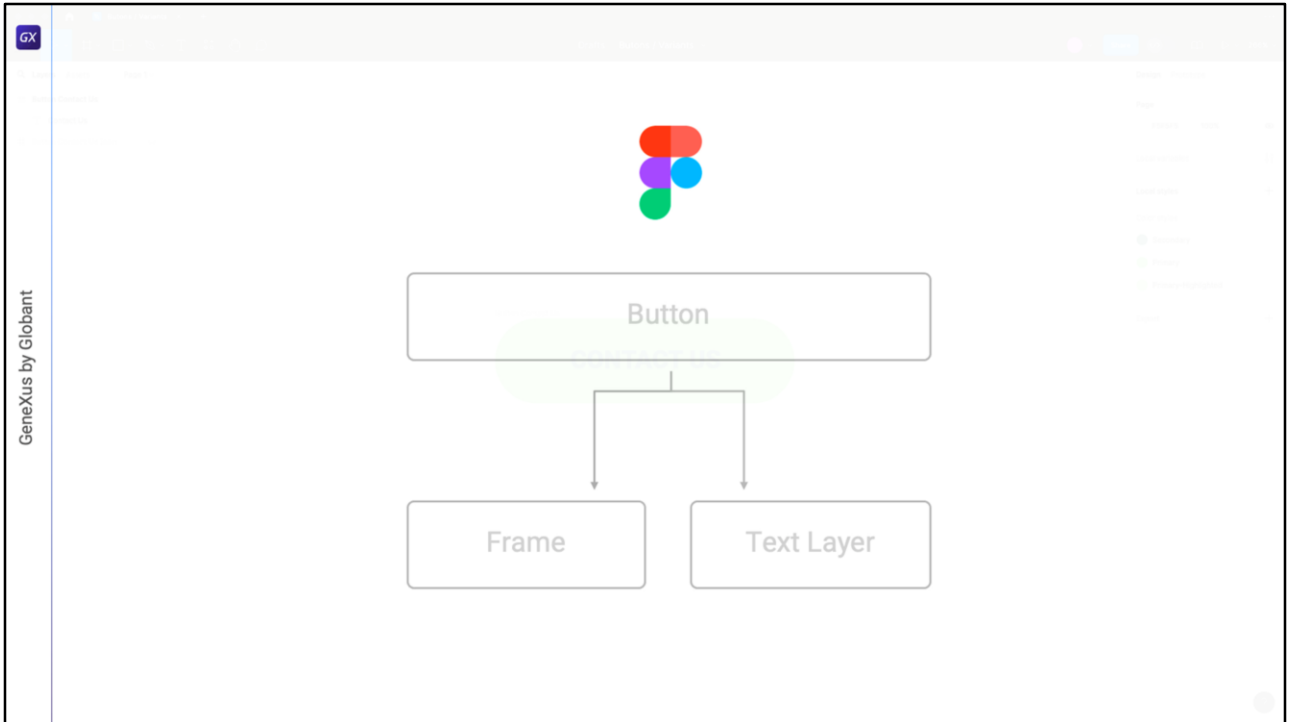


Buttons in Figma

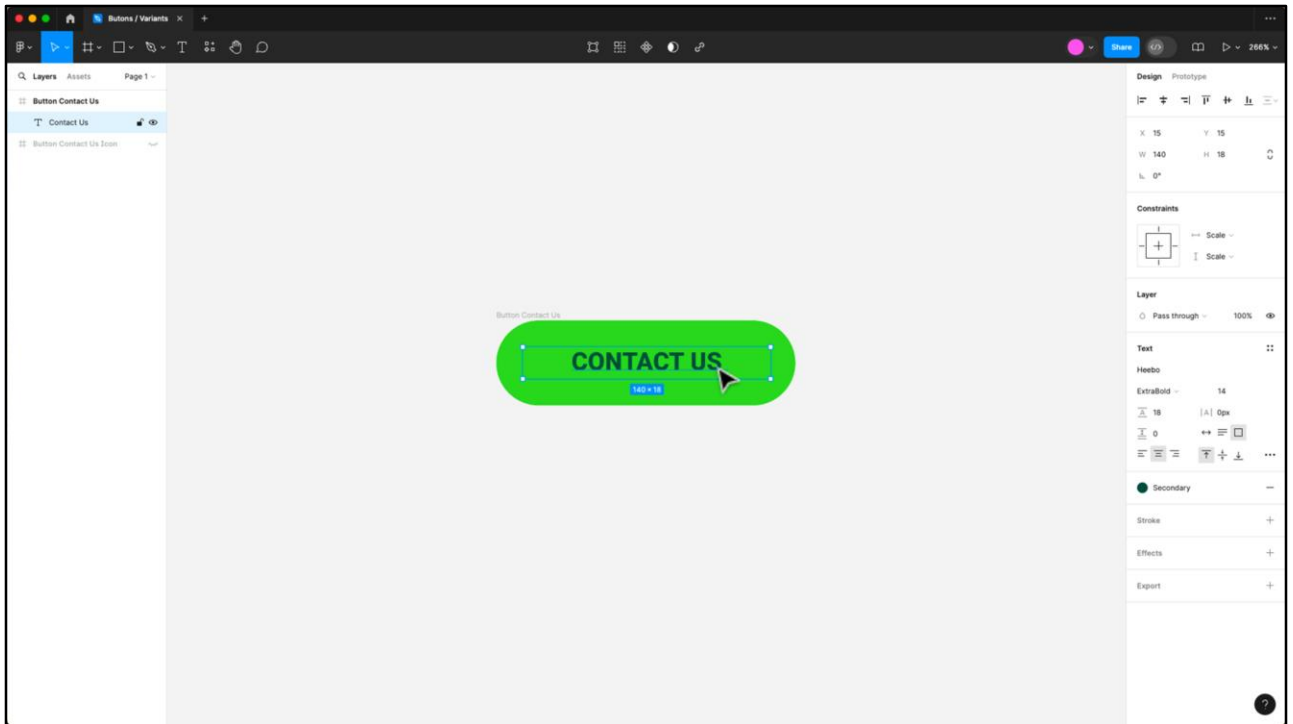


Cecilia Passalacqua

In this section, we will be talking about buttons. What I have to take into account to build them in Figma, and how to generate their variants.

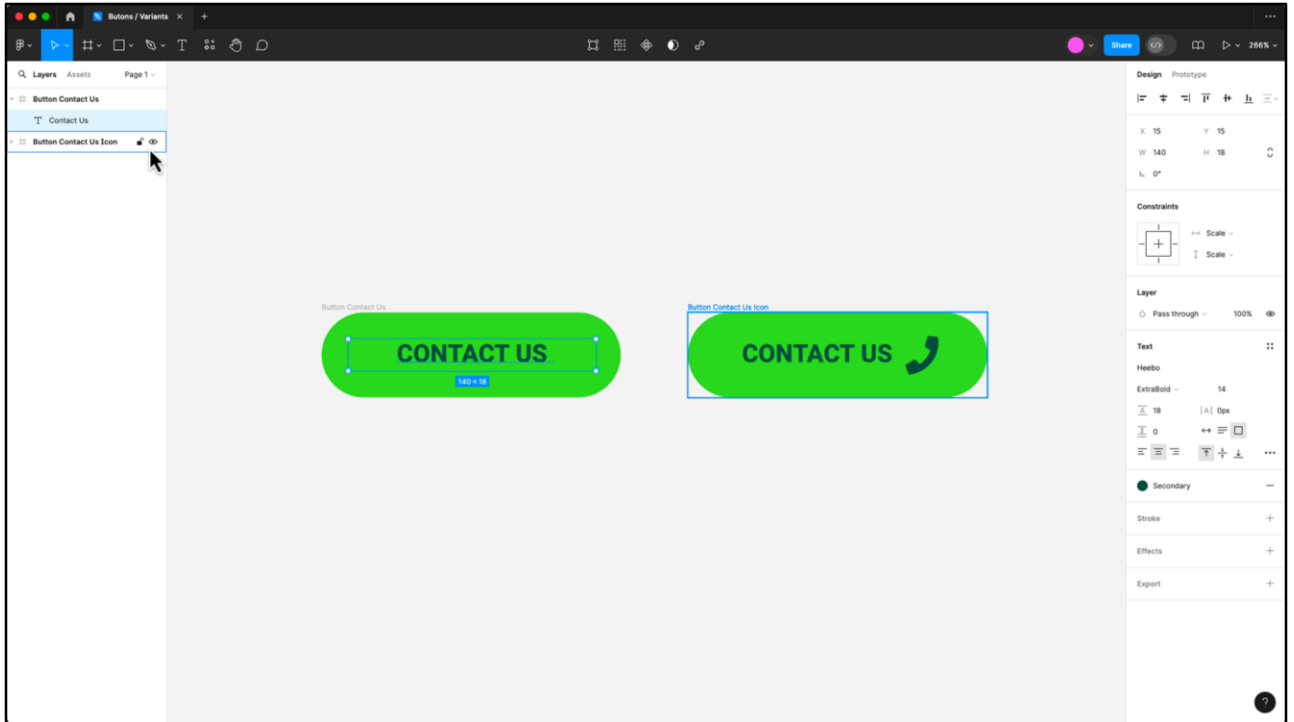


Let's start from this concept: in Figma there is no "button" element as such; it is built from 2 elements, a frame container and a text layer.

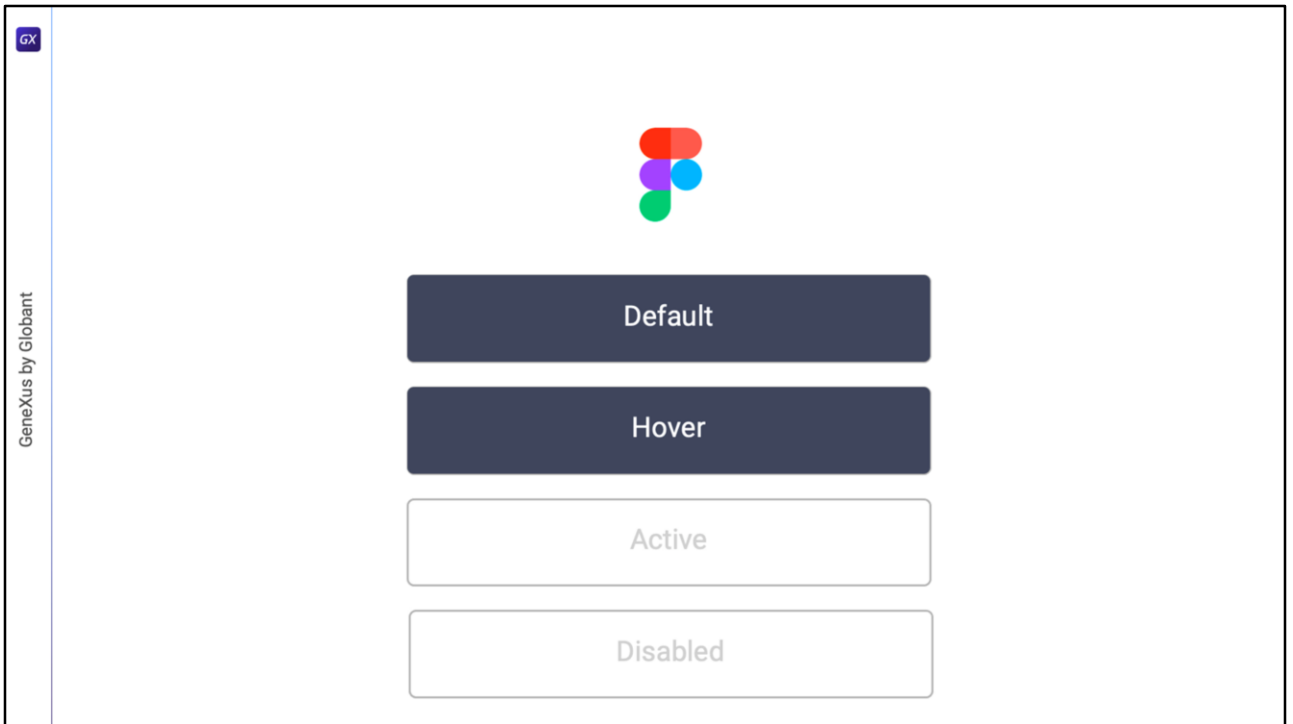


As we saw in previous classes, the frame element is a container that can have properties such as height, width, rounded edges, background and stroke.

To this frame we add a text layer that contains the caption of the button.

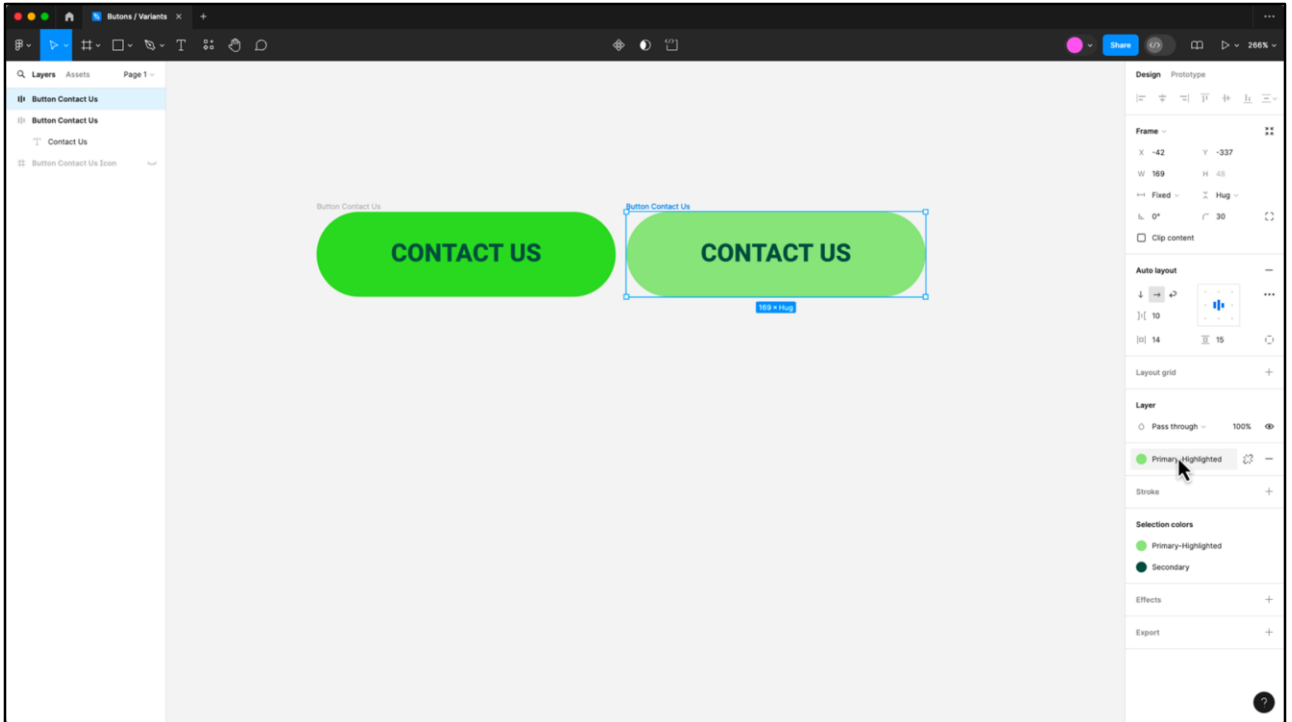


We could also add an icon as in this example, but this is not the case we are using for our application.



We often say that buttons have at least 4 states: default, hover, active and disabled.

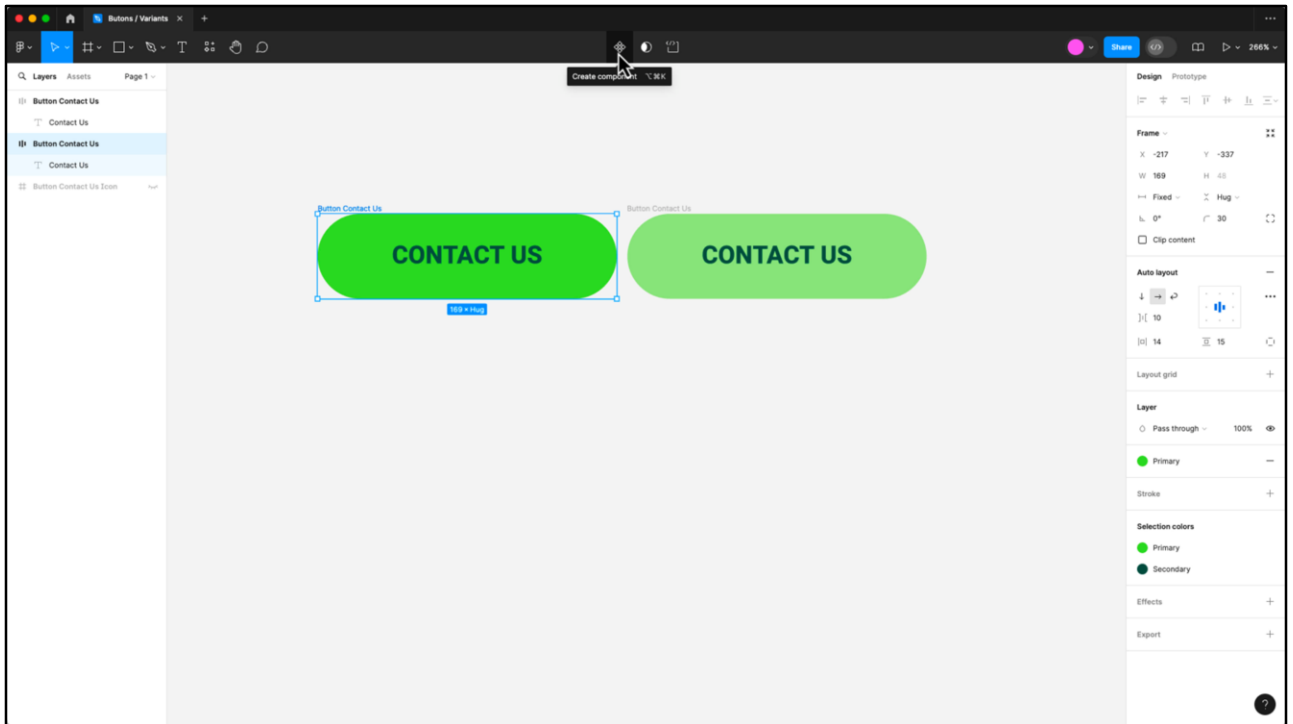
In this case, we are only going to make variants for the default and hover states.



Once I decide how the button will be displayed in the default state, I duplicate it and generate the hover variant. In this case, the only thing that changes is the background-color from Primary to Primary-Highlighted, a color style that I had previously defined.

Among the design decisions I'm making at this point, including colors, fonts, and sizes, I can also include the settings we saw in previous classes. What is going to be the behavior of the button, for example, if the text changes and becomes longer? Do you remember those options?

In this case, I'm going to leave it fixed. It is not the best way to model it, but in the next classes with Ceci you will see how to add these behaviors directly in the GeneXus button component.

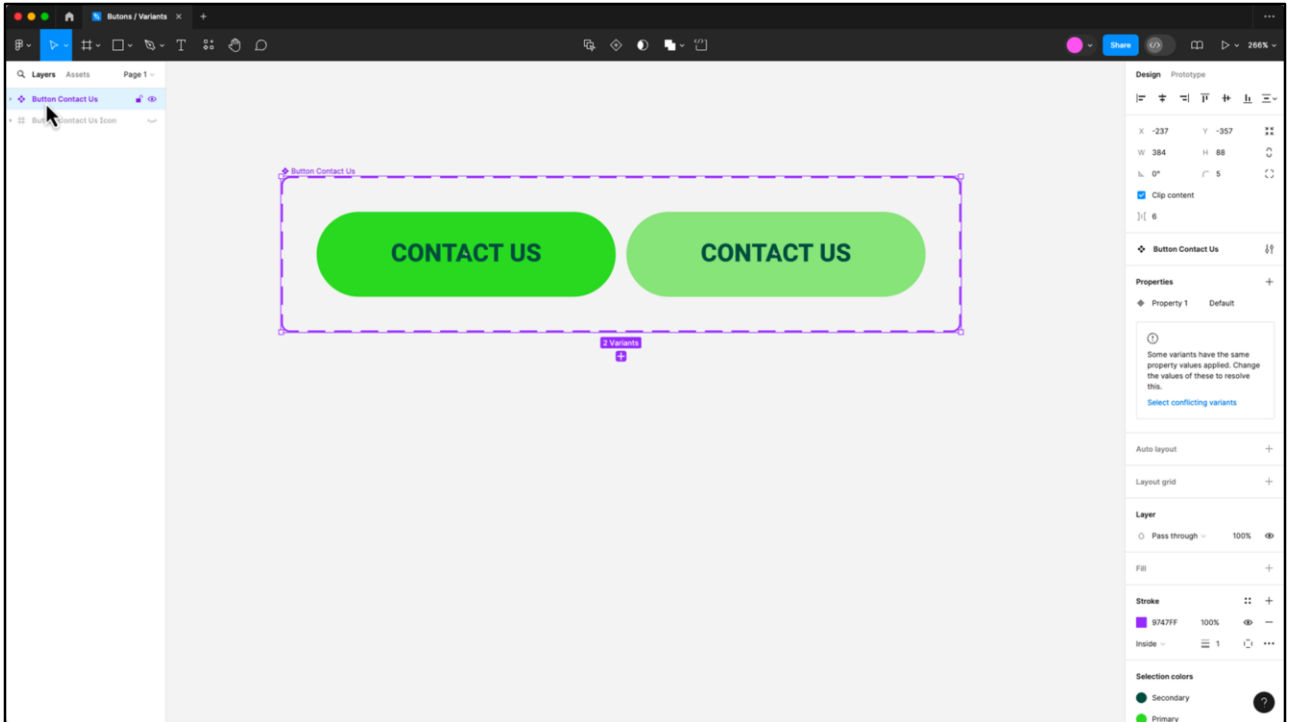


Let's go back to our two buttons.

Since we know that the button is going to be used in several instances, we are going to generate it as a component, and it is important that each state of the button is a different component.

Unlike this, which is a component and an instance of a component.

By having them modeled as 2 different components, we can combine them as variants...



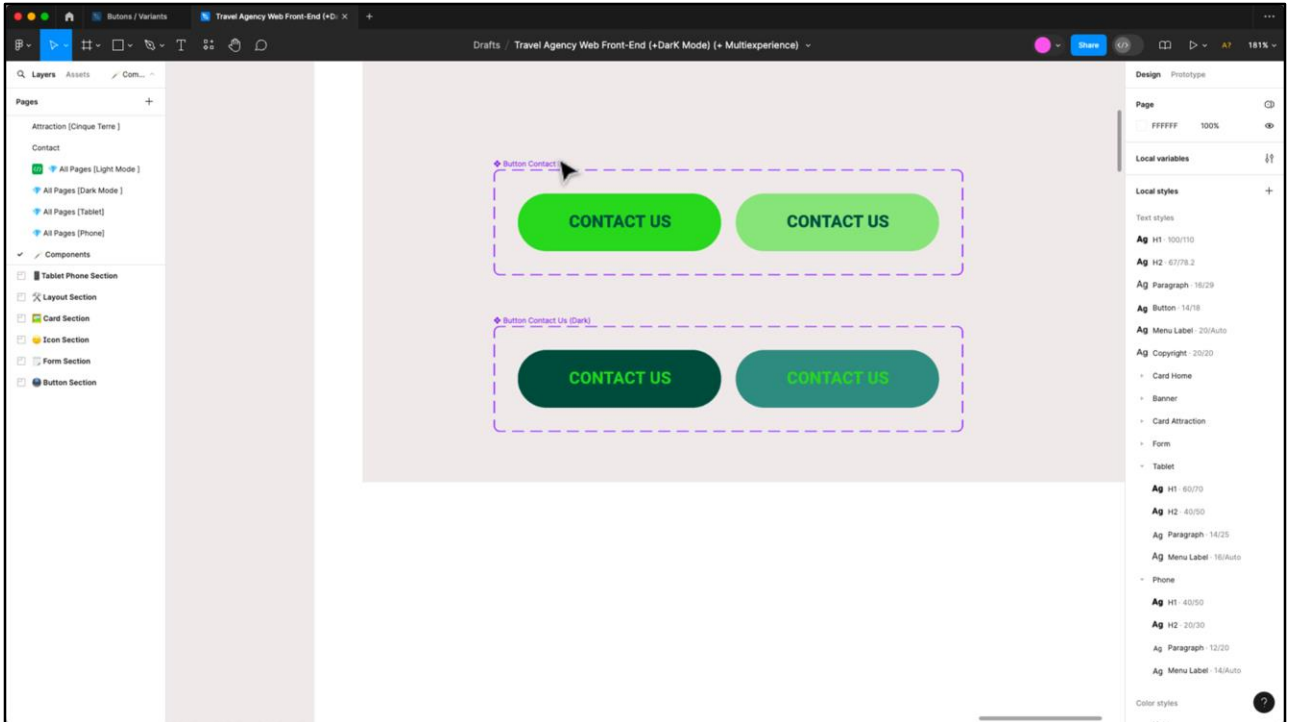
...selecting them and choosing in the column on the right the option "Combine as variants".

Both components are left inside the same one, and if we look at their layer names they are the same; they are called "Default".

For the sake of consistency and to avoid confusing the frontender, let's change these names. We double-click on the layer and replace "property 1" with "Button Contact", and on the other side of the equal sign, with the corresponding state.

In this way, the variants will be called Button Contact=Default and Button Contact=Hover.

When we use an instance of these variants, we will be able to quickly select from the column on the right which state we want to represent.



Note that for the Dark Mode case I made another set of variants, but I could have just as well made all 4 as variants of the same button.

In the next class with Ceci, you will see how to take this information from the design file and take it to a button type control in GeneXus.

GX

GeneXus by Globant

GeneXus[™]
by **Globant**

training.genexus.com