There are many different platforms for developing mobile apps that require either limited or no coding. This is usually done by using pre-developed objects known as components, coupled with some block programming.

The Platform we will be using for this activity is called **Thunkable**. This service has a free (public) app development section and a paid (private) app development section. For this camp we will be utilizing the free or public app development tier. If you have an Android or iOS mobile device you can download their live app viewer app from the respective app store. If you do not have a mobile device do not worry as you can live preview your app directly in the web browser for most functions.

We will be developing a mobile app to work as a Translation App in both text and verbal formats this can be a useful tool to translate things you may come across on the internet or when trying to work with people from different countries on a project which demonstrates the concept of **Availability**. The layout should be easy to use for everyone utilizing the concept of **Keep It Simple**.

**Section: Account Creation / Access:**

If you have not created an account yet please go to: <https://x.thunkable.com/signup> , if you have created an account you can skip this section.

You will need to enter your email address into the box with the placeholder text “Enter your email” and then click the button “Email me the link”.

Check your email and you should receive a link to click which will log you into the site on your device. If you do not click the logout button afterwards it should keep you logged in for 30 days and you can just revisit the site to continue working.

**Section: Account Access:**

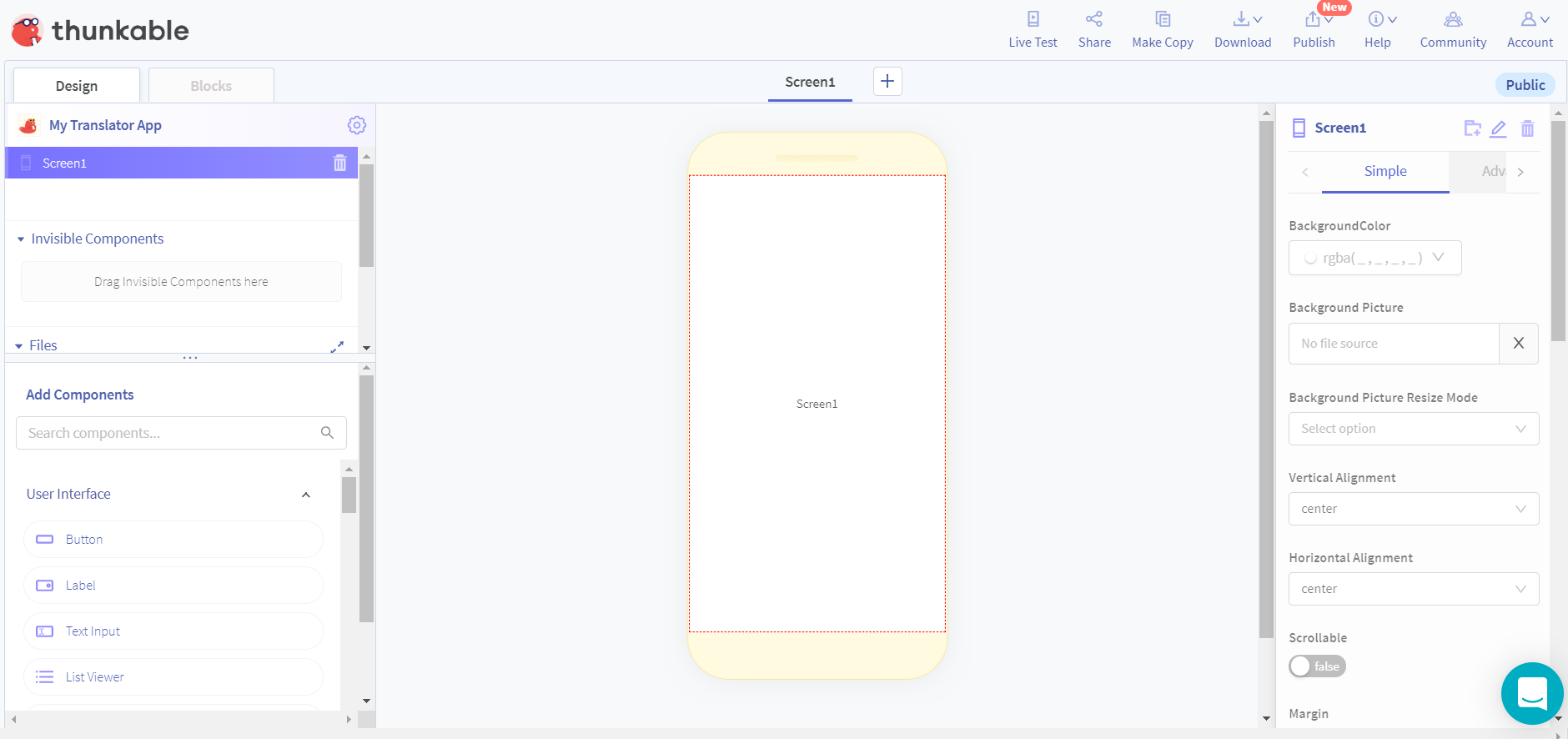
If when you visit the site again you find yourself logged out you will need to go to <https://x.thunkable.com/login> and once again provide your email address and click “Email me the link”. Then log into your email and click the link to be automatically logged into the site.

**Section: Getting Started:**

|  |  |  |
| --- | --- | --- |
| 1. Click the Start Building Button | 1. Fill in the Project Name “My Translator App” and then select a relevant category such as “Education” | |
| 1. Make sure the checkbox for “Use the Drag and Drop builder” is checked. 2. Then Click the “Create” button 3. Hide the tutorials section by clicking the purple tab with the left arrow on it. | |  |

**Section: The Design Process:**

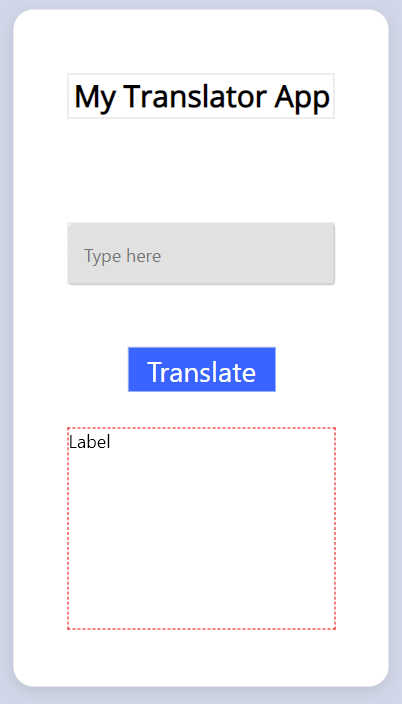
1. At this point, your screen should look like the one below:



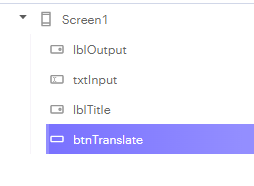
On the left at the top you will see 2 tabs, Design and Blocks. To begin we will be working on the design of the app so make sure that tab is selected. Below that is a section for components that are currently part of your app and further below that are components you can drag onto your app.

On the right will be the individual properties of whatever the current selected component is. To begin it is showing the properties of the Screen1 component.

1. This program is going to consist of 4 main components: A Translate **Button**, Application Title **Label**, and Output **Label** and **Text Input** component to be translated. To begin drag one of each component onto your “phone” screen in the middle. Your screen should look like the image below but rotated up and down instead of laying on its side as depicted here although the different elements will be in their default state:



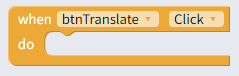
1. Arrange the 4 items so that the input element is about 80% of the width of the device, center the Label and Button elements. Finally make the output label take up the remaining space. You can re-adjust the size of these as you fill in the next steps.
2. Now we need to adjust the Label. To begin, click the label you dragged into the center of the screen. Then looking on the panel on the right side, adjust the following options
   1. Text: My Translator App
   2. Font Size: 30
   3. Font Weight: Bold
   4. Text Align: Center
3. Finally, the Button should be selected. Change the Text from “Button” to “Translate”. Apply any visual changes you would like. In this example the font Size was adjusted to 28 and the radius was removed.
4. Now on the left-hand side of the browser you should see a heading called Screens with each one of the applications components listed below it. Rename each component so it makes more sense than their default names. We will prefix each item with a 3-letter component designator followed by a name that describes its function. Label1 should be renamed “lblTitle”, Label2 should be renamed to “lblOutput”, Text\_Input1 should be renamed “txtInput”, and finally the button “Button1” should be changed to “btnTranslate”



At this point the Design Process is finished. Of course, this app is not visually stunning, but you can spend as much time as you like working on layouts and design aspects of a specific app when you are developing them on your own. Now we will move into the block coding of this app to provide the functionality.

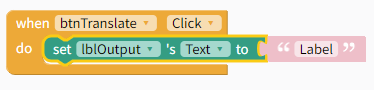
**Section: Block Coding:**

1. To begin, click on the Blocks tab mentioned earlier. Then you should see “btnTranslate” in the list of items on the left. Clicking on this opens some items associated with the button component. The one we want should be at the top for when Button1 is clicked do something. Drag this to our screen on the right.



1. Next select the lblOutput. Then select the following item and connect it to the DO section of your previous block.





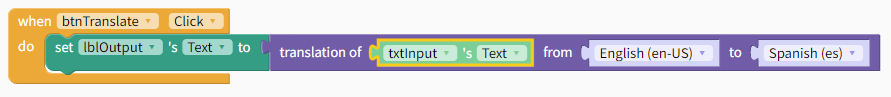
1. Next select the Speech under App Features and select the below item from the list. This is going to be connected to the previous set block on the right hand side, replacing the “Label” block. 

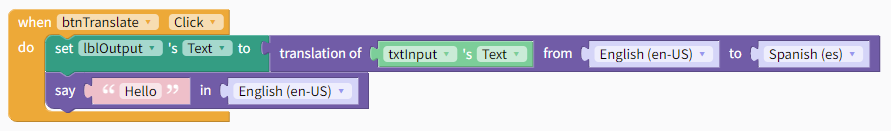


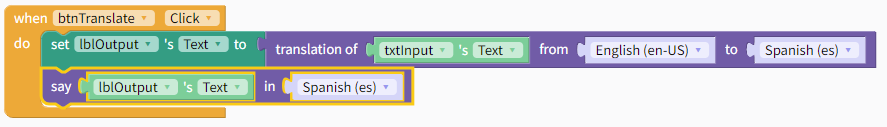
1. Next click on the txtInput item from the left-hand side and select the block that says “txtInput’s Text.



Place this block in the first spot of the Translation code piece, replacing “Hello”.



1. The coding is almost finished. Click back on Speech under App Features and select the Say “Hello” in English block. Add this block below the current Set block but still inside the DO section of the When block. 
2. To customize the speech part, we need to now click on the lblOutput component on the left hand side again and this time select lblOutput’s Text block. Place this block where the word “Hello” is in the previous say block. Finally click the dropdown for English and select Spanish(es) form the options.



1. Now you can click the “Live Test” icon at the top of the screen. If you have the Thunkable live test app on a mobile device you can run it there, otherwise it can be tested directly in the browser window as well. After clicking the Live Test icon, a window should pop up and on the left is the live version of your app. Test by entering a word and then click the translate button.

**Customize:**

1. Change the colors and layout of the app to be unique. Remember to keep the layout Simple as possible to allow ease of use for anyone. Add a background image appropriate for the city in which you reside. Make sure it fits correctly, you may need to adjust the image or crop it in an editor first.

**Section: Team Project**

Play around with the features and components and try to develop some sort of basic app to show off to the camp. Keep the GenCyber concepts in mind and use at least one of them in the design/concept process of the basic app.

Must use a minimum of 3 components and be demonstrable to the camp.