

# MAIN TITLE SLIDE HERE

---

Remember Girl Scouts of Orange County reference and authored by a “volunteer” in the event this content is used by other councils.

Restate the prerequisites? a) need to know a gmail account un/pw  
b) need to know how to connect your phone to the event WiFi c)  
need to know how to load the app “Thunable Live” from the phone’s app store

Link to “remix” project:

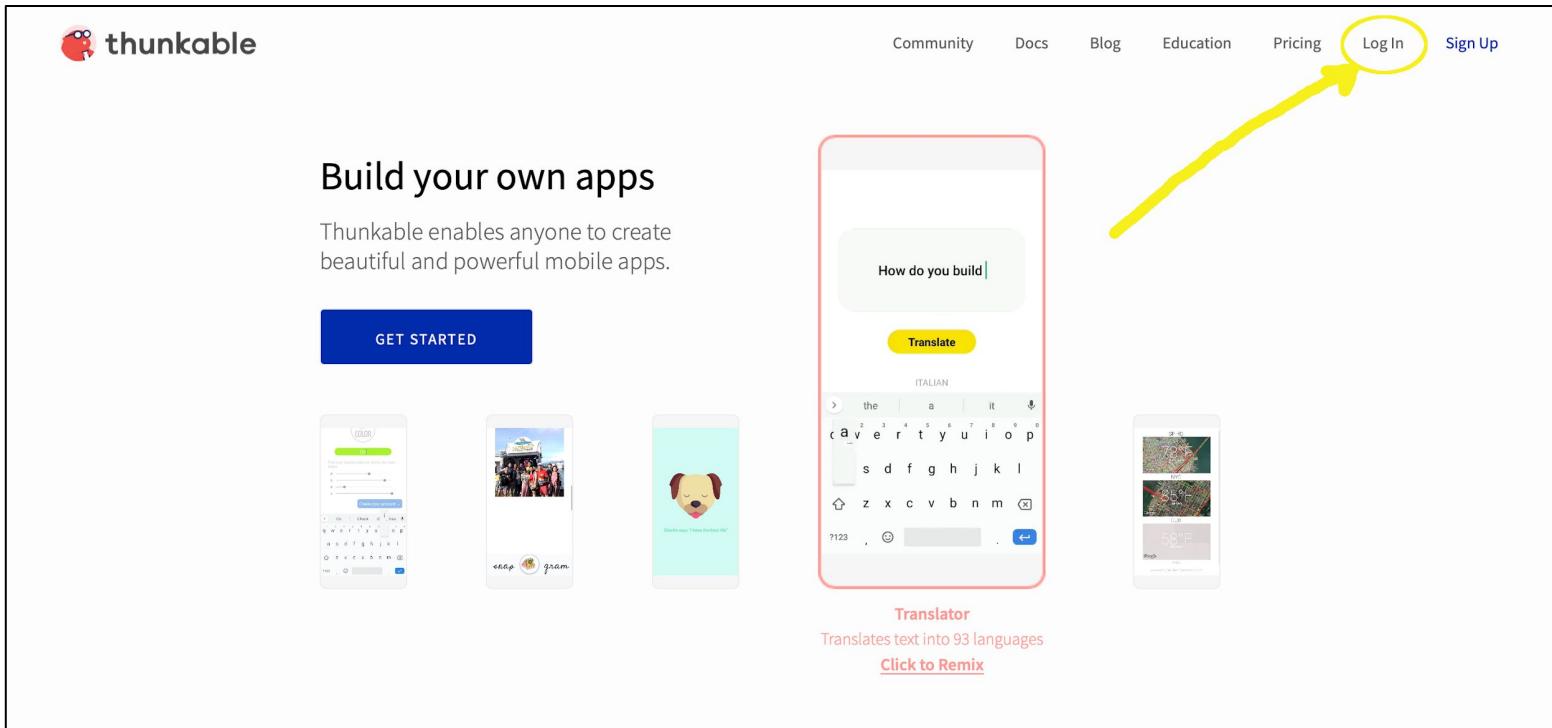
<https://x.thunkable.com/copy/478ba55b51ae0c917a632b0955aa454>  
1

Might need to provide app icon as a downloadable link, seems like you need to browse that in.

As a backup, have the assets stored in a GITHUB repository. We could use GITHUB to hold links to the files above and URL for the remix too

# Getting Started - ThunkableX

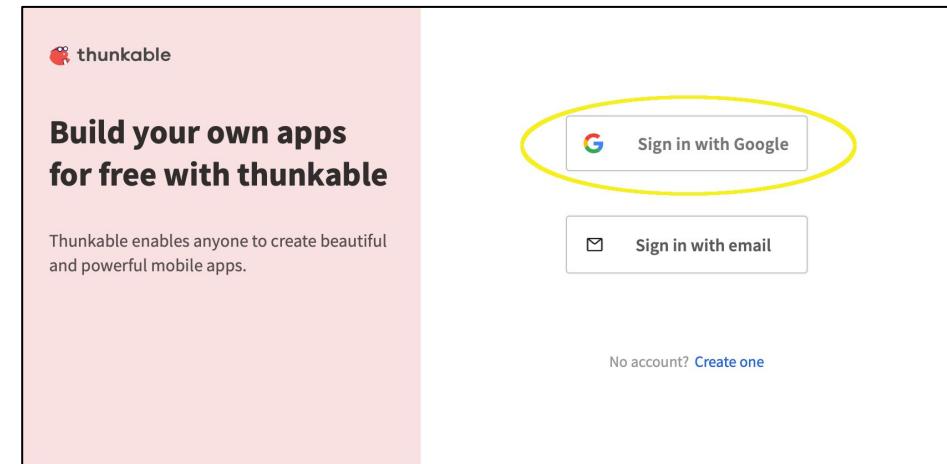
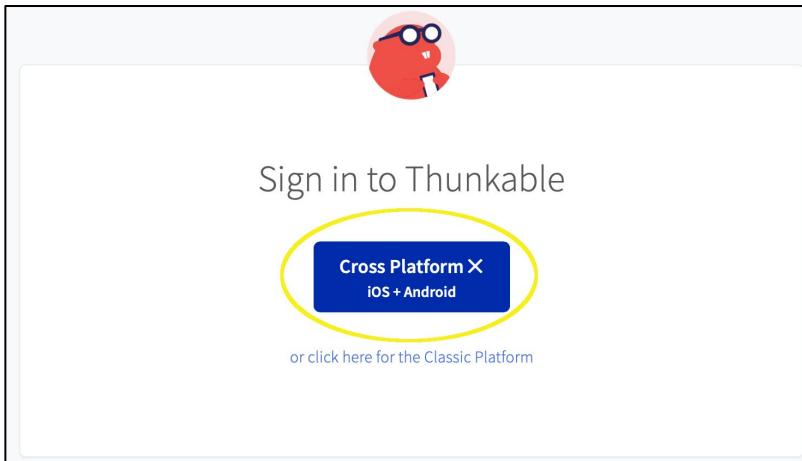
Using the **FIREFOX** browser on your laptop, navigate to the <https://www.thunkable.com> website and click the Login button in the upper right corner of the screen as shown below.



The screenshot shows the Thunkable website homepage. At the top, there is a navigation bar with links for Community, Docs, Blog, Education, Pricing, Log In (which is circled in yellow), and Sign Up. Below the navigation bar, the text "Build your own apps" is displayed, followed by the subtext "Thunkable enables anyone to create beautiful and powerful mobile apps." A large blue "GET STARTED" button is visible. On the right side of the page, there is a section titled "Translator" which includes a screenshot of a mobile app interface showing a keyboard and text input field, with the caption "Translates text into 93 languages" and a "Click to Remix" link. A yellow arrow points from the bottom left towards the "Log In" button in the navigation bar.

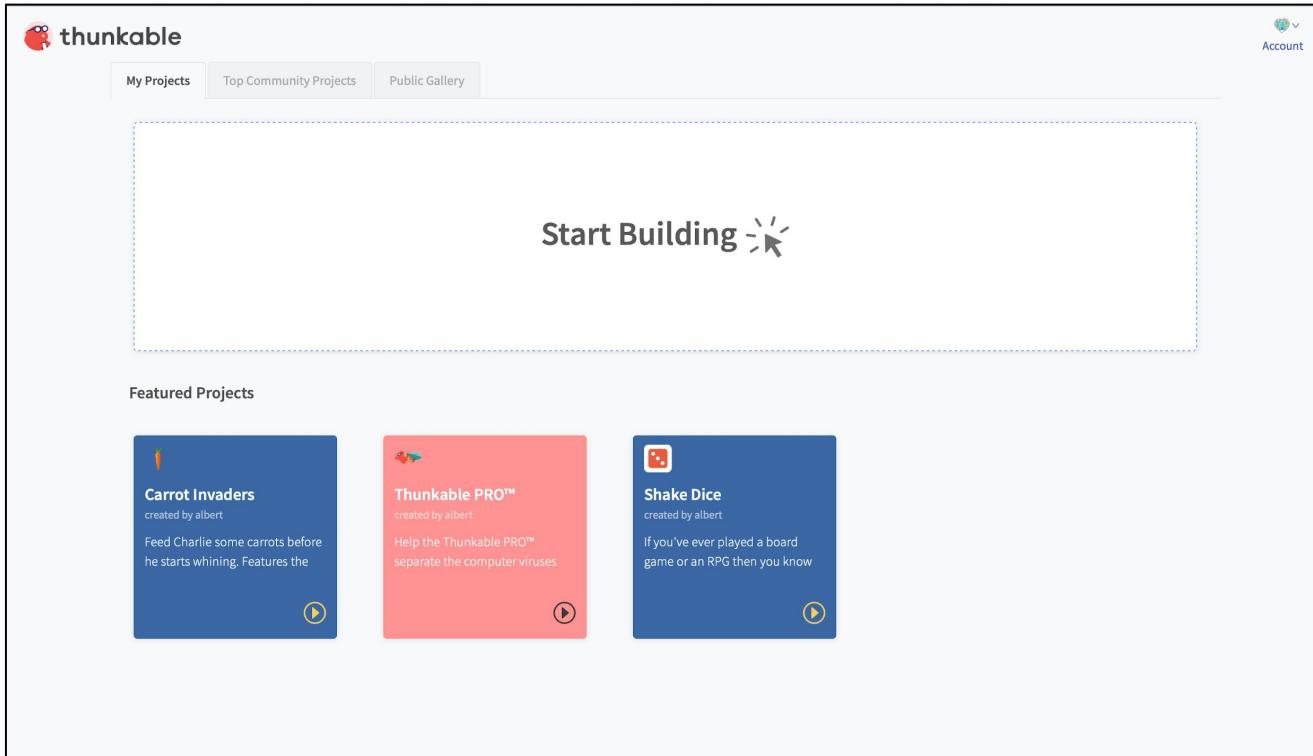
# Getting Started - ThunkableX

Click on the Cross Platform X button, and then click on the Sign in with Google button. **Use the GMAIL username and password you were asked to bring to this event to log in.**



# Getting Started - ThunkableX

Now that you are logged on, we'll start by REMIXing an existing application.



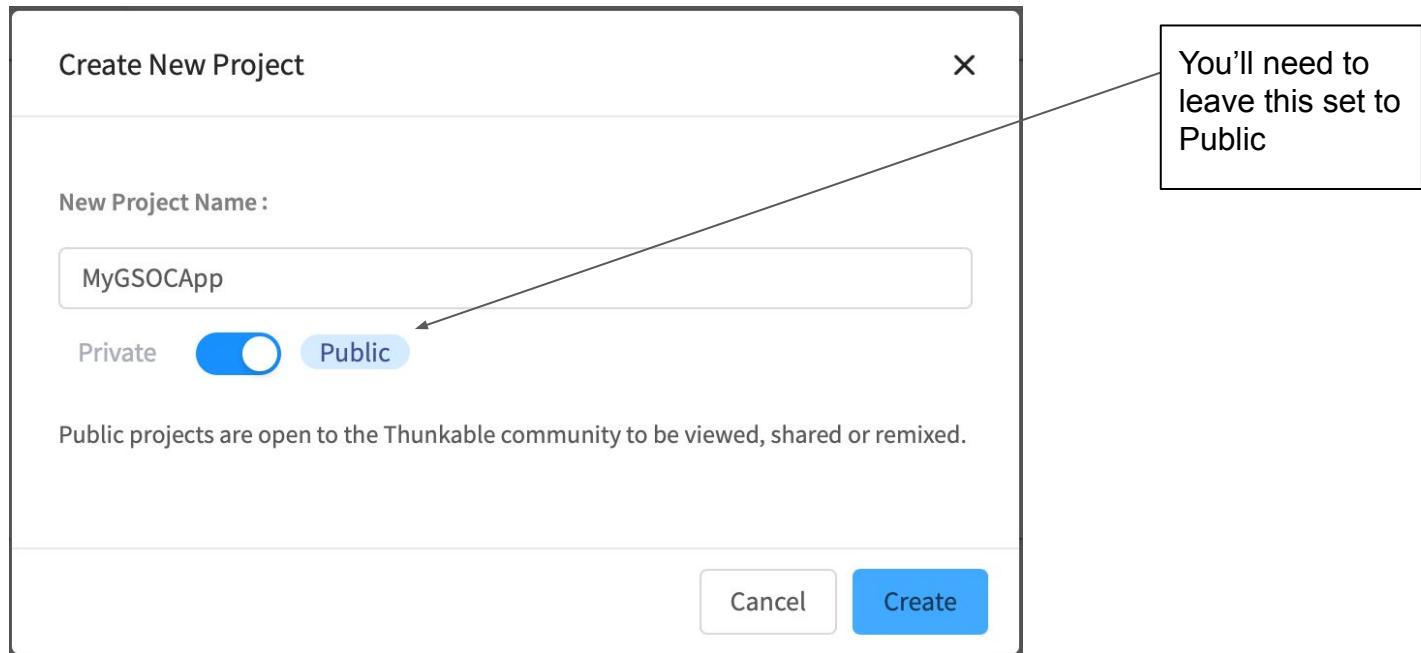
The screenshot shows the ThunkableX web interface. At the top, there is a navigation bar with the Thunkable logo, account information, and tabs for "My Projects", "Top Community Projects", and "Public Gallery". Below the navigation, a large central area features a "Start Building" button with a cursor icon pointing at it. This area is surrounded by a dashed blue border. Below this, under the heading "Featured Projects", are three project cards:

- Carrot Invaders** (created by albert): Feed Charlie some carrots before he starts whining. Features the
- Thunkable PRO™** (created by albert): Help the Thunkable PRO™ separate the computer viruses
- Shake Dice** (created by albert): If you've ever played a board game or an RPG then you know

Each project card includes a play button icon in the bottom right corner.

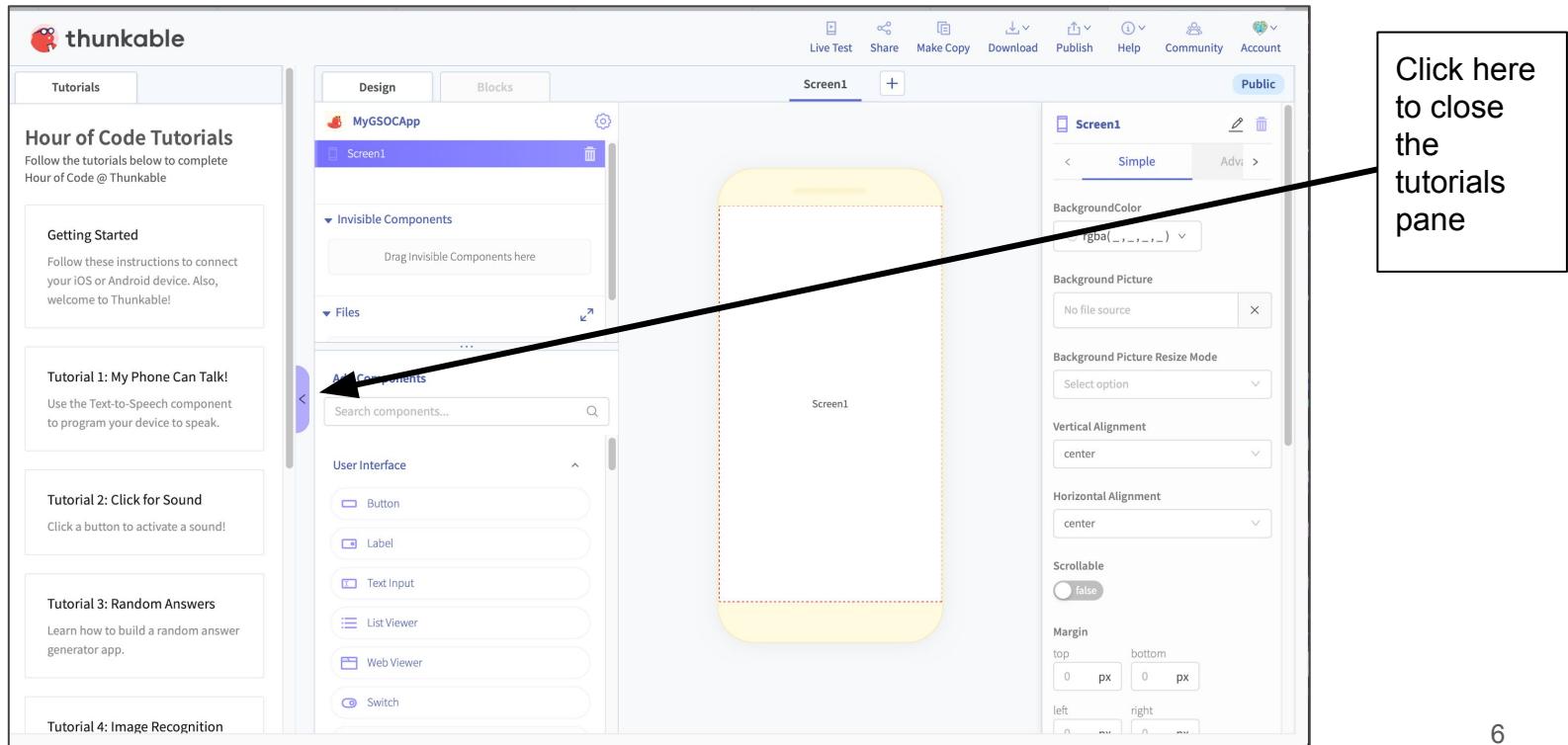
# Getting Started - ThunkableX

Let's name our project "MyGSOCApp" as shown below:



# Getting Started - ThunkableX

**Congratulations!!!** Let's get started! Note that you can access TUTORIALS using the left side pane, but we'll close the tutorial pane for the remainder of our event.



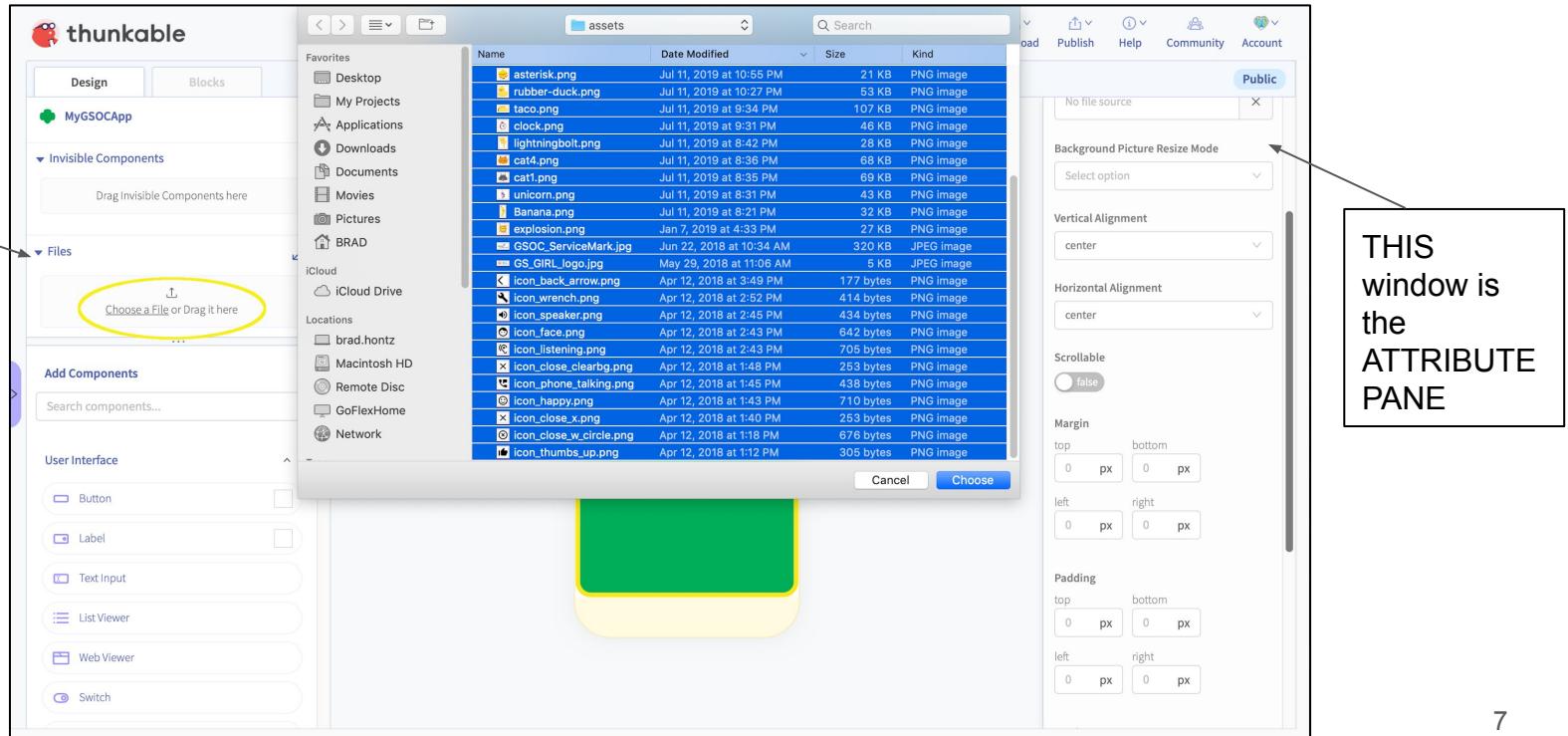
The screenshot shows the ThunkableX interface. On the left, there is a sidebar titled "Hour of Code Tutorials" containing five sections: "Getting Started", "Tutorial 1: My Phone Can Talk!", "Tutorial 2: Click for Sound", "Tutorial 3: Random Answers", and "Tutorial 4: Image Recognition". On the right, the main workspace shows a "Design" tab selected. A mobile phone component is placed on the canvas. The right panel contains various settings for the phone component, such as "Background Color" (set to "rgba(255, 255, 255, 1)"), "Background Picture" (set to "No file source"), "Background Picture Resize Mode" (set to "Select option"), "Vertical Alignment" (set to "center"), "Horizontal Alignment" (set to "center"), "Scrollable" (set to "false"), and "Margin" (set to "top: 0px, bottom: 0px, left: 0px, right: 0px"). A black arrow points from the "Tutorials" section in the sidebar to the close button in the top right corner of the tutorials pane.

Click here to close the tutorials pane

# Getting Started - importing ASSETS we'll use

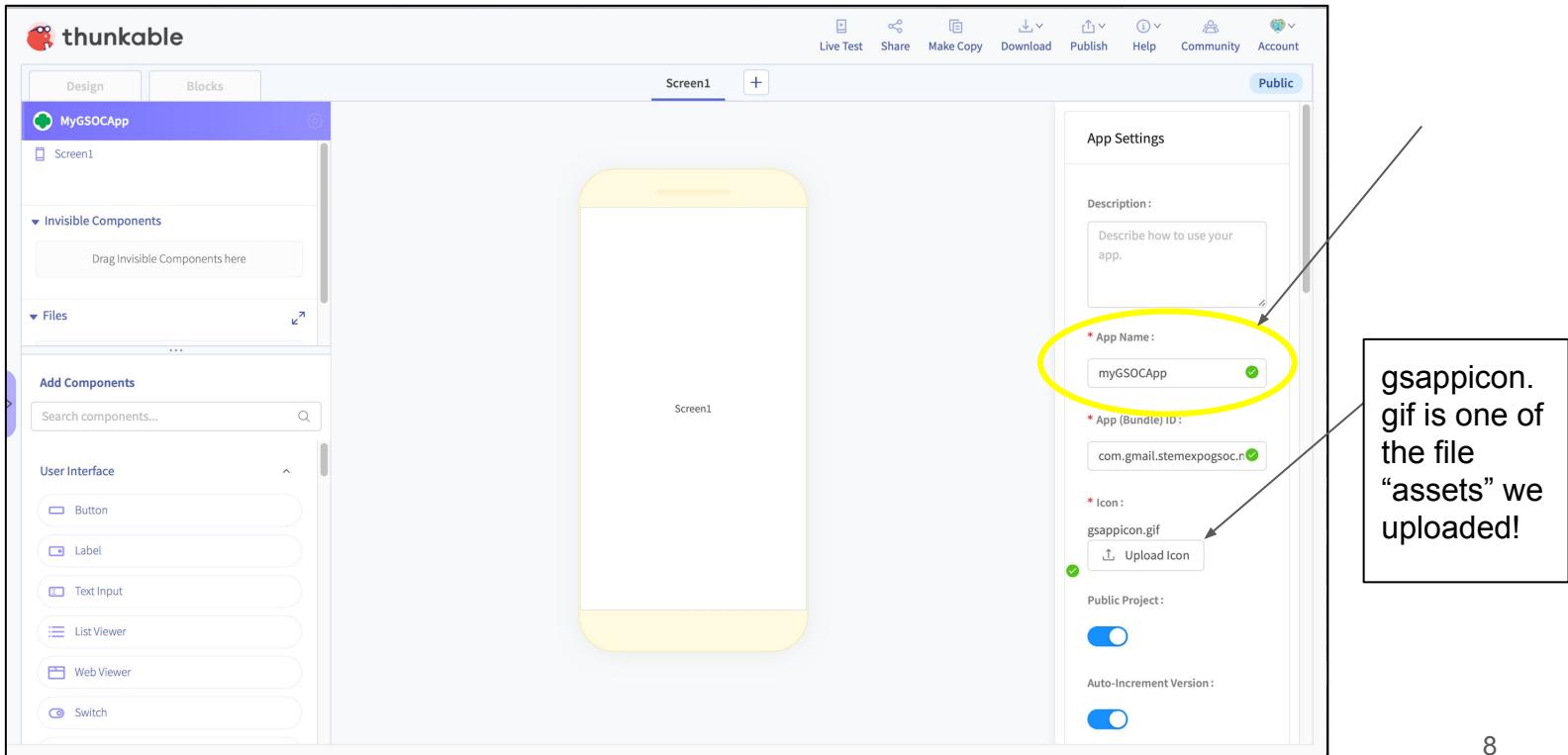
App developers use digital “assets”; files containing images, icons, sounds, html content, etc. to build their apps. As shown below, import all files from the folder:

C:\Documents\Events\CodingForGoodApp\assets



# Screens, Components and Blocks!

Let's talk about App Settings, Screens, Components, and Blocks! Use the ATTRIBUTE PANE on the right to name our app and assign our app an icon (as seen on our phone).



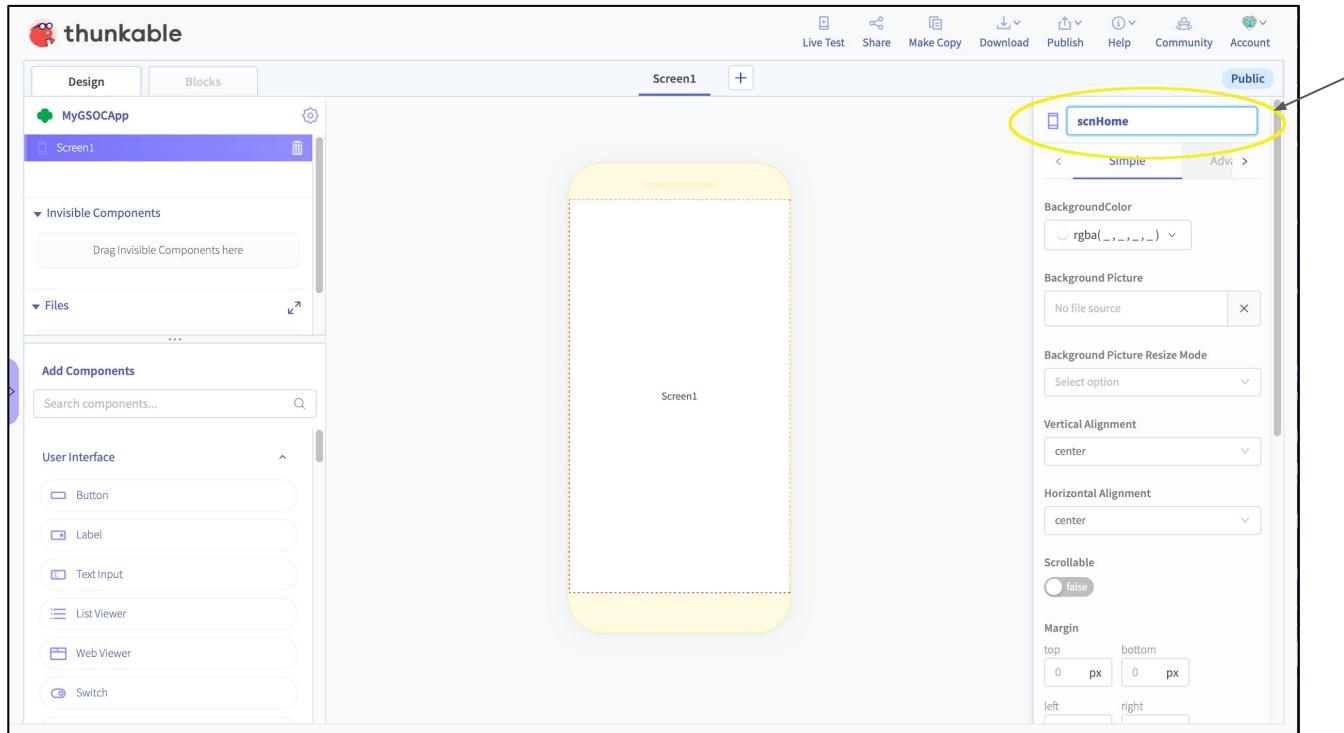
The screenshot shows the Thunkable app builder interface. On the left, the project navigation bar shows 'MyGSOCApp' with 'Screen1' selected. The main workspace is titled 'Screen1'. On the right, the 'App Settings' pane is open, showing the following fields:

- Description: Describe how to use your app.
- \* App Name: myGSOCApp
- \* App (Bundle) ID: com.gmail.stemexpogsoc.n
- \* Icon: gsappicon.gif (with an 'Upload Icon' button)
- Public Project: (switch is on)
- Auto-Increment Version: (switch is on)

A yellow circle highlights the 'App Name' field, and an arrow points from it to a callout box containing the text: "gsappicon.gif is one of the file 'assets' we uploaded!"

# Naming conventions we'll use

The following slide will contain a list of naming conventions that we'll use for this project. App developers use conventions like this to make it easier to understand the screens and components they are referencing within their code.



# Naming Conventions - Reference

---

You will find it much easier to code your app when using meaningful PREFIX and names to identify your app's screens and components:

**scn** = Screen (e.g. scnHome, scnLaw, scnPromise ...)

**btn** = Button (e.g. btnHomeLaw, btnLawReturn, btnPromiseSpeak ...)

**lst** = List (e.g. lstTroopMembers)

**row** = Row layout container component (e.g. rowHomeBotttom)

**col** = Column layout container component (e.g. colHomeTop)

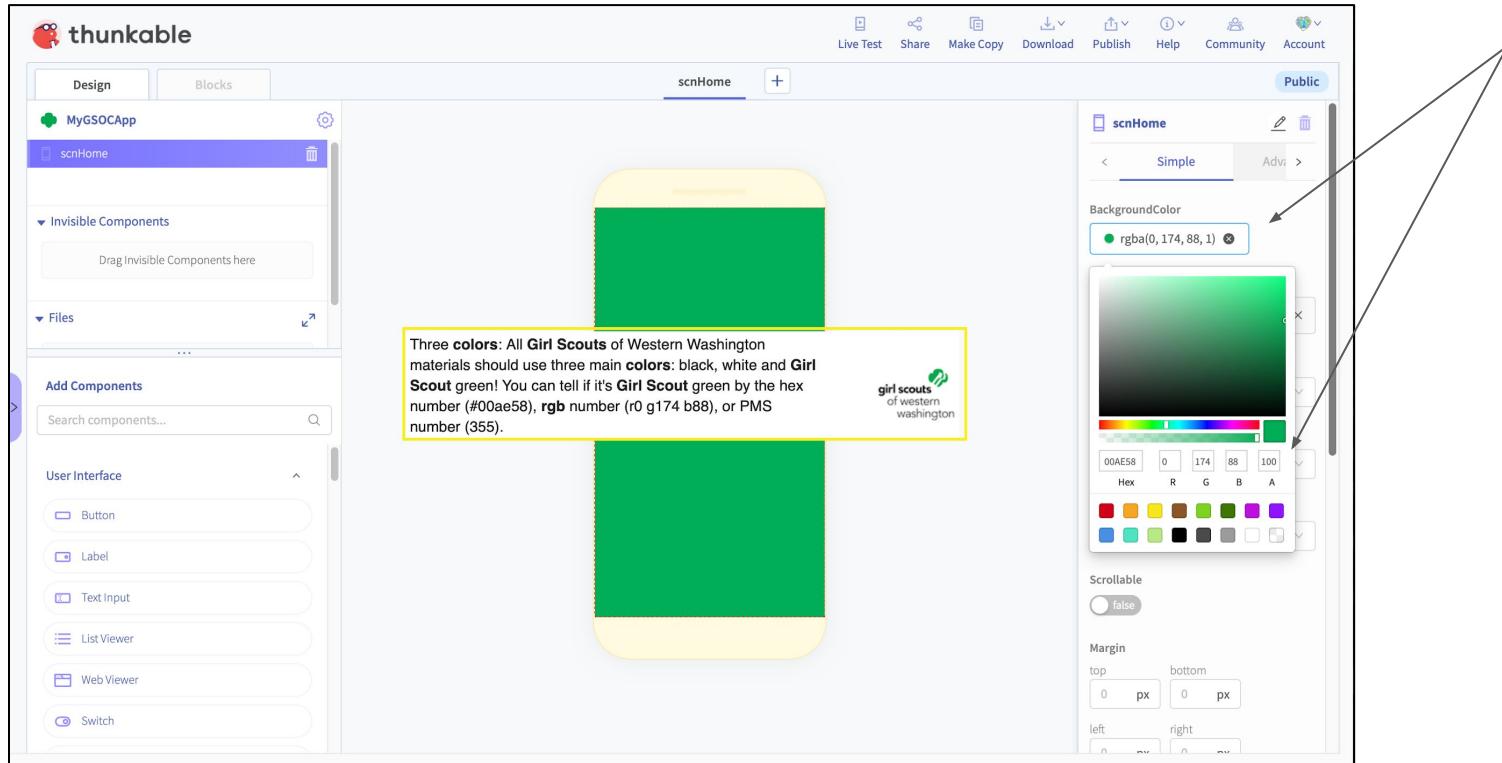
**snd** = Sound file object (e.g. sndPromise, sndLaw)

**img** = Image file object (e.g. imgGSOCLogo)

**wv** = “Web Viewer” (e.g. wvPromise)

# Home Screen Background Color

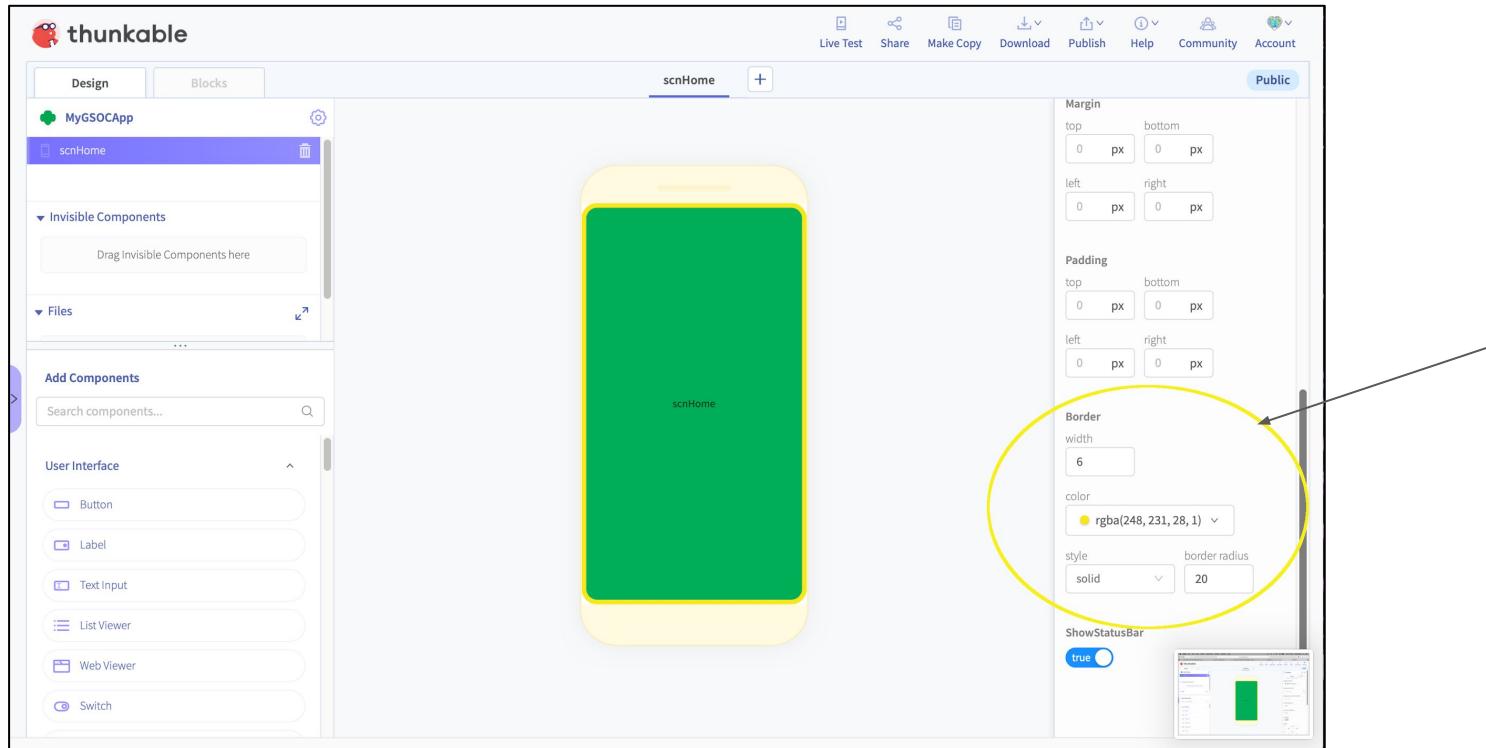
Let's use Girl Scout colors for our home screen background! The RGB(A) representation of "Girl Scout Green" is R = 0, G = 174, G = 88. Set our background as shown below:



The screenshot shows the Thunkable app interface. On the left, the sidebar includes 'Design' and 'Blocks' tabs, a project title 'MyGSOCApp', and a screen list with 'scnHome' selected. Below these are sections for 'Invisible Components' and 'Files'. Under 'Add Components', there's a search bar and categories for 'User Interface' (Button, Label, Text Input, List Viewer, Web Viewer, Switch), 'Data', 'Control', 'Math', 'String', 'Color', 'Image', and 'File'. The main workspace shows a yellow rounded rectangle containing a green square. A callout box over this area contains the text: 'Three colors: All Girl Scouts of Western Washington materials should use three main colors: black, white and Girl Scout green! You can tell if it's Girl Scout green by the hex number (#00ae58), rgb number (r0 g174 b88), or PMS number (355).'. To the right of the workspace is the screen configuration panel for 'scnHome'. It shows the screen name 'scnHome' and its preview. The 'BackgroundColor' setting is set to 'rgba(0, 174, 88, 1)'. The color picker shows a gradient from black to green, with the green color highlighted. Other settings in the panel include 'Simple' (selected), 'Advi', 'Scrollable' (false), 'Margin' (top: 0px, bottom: 0px, left: 0px, right: 0px), and a color palette.

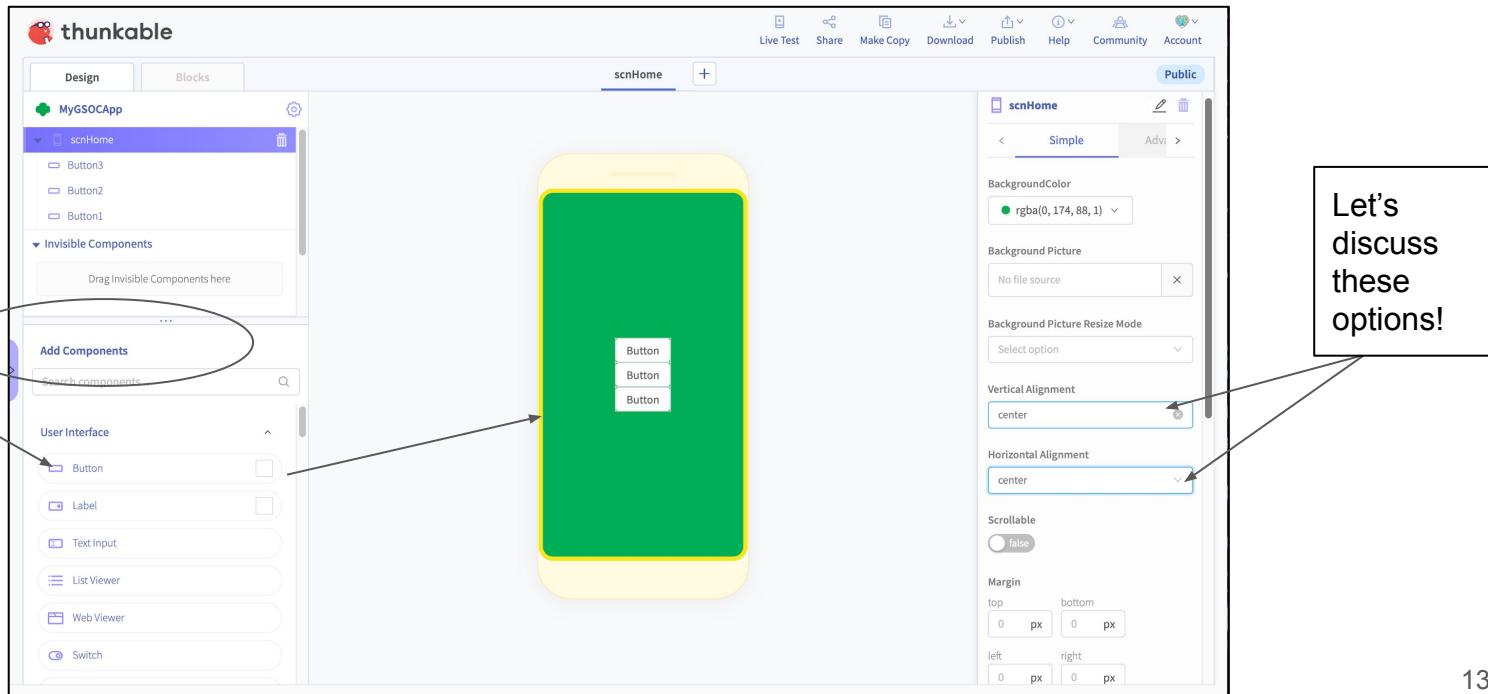
# Home Screen: Set a Border

A border might look cool on our home screen. Scroll the scnHome ATTRIBUTE WINDOW down a bit and set the border width, color, and radius. Radius determines how round the border is.



# So what is “Layout” anyway?

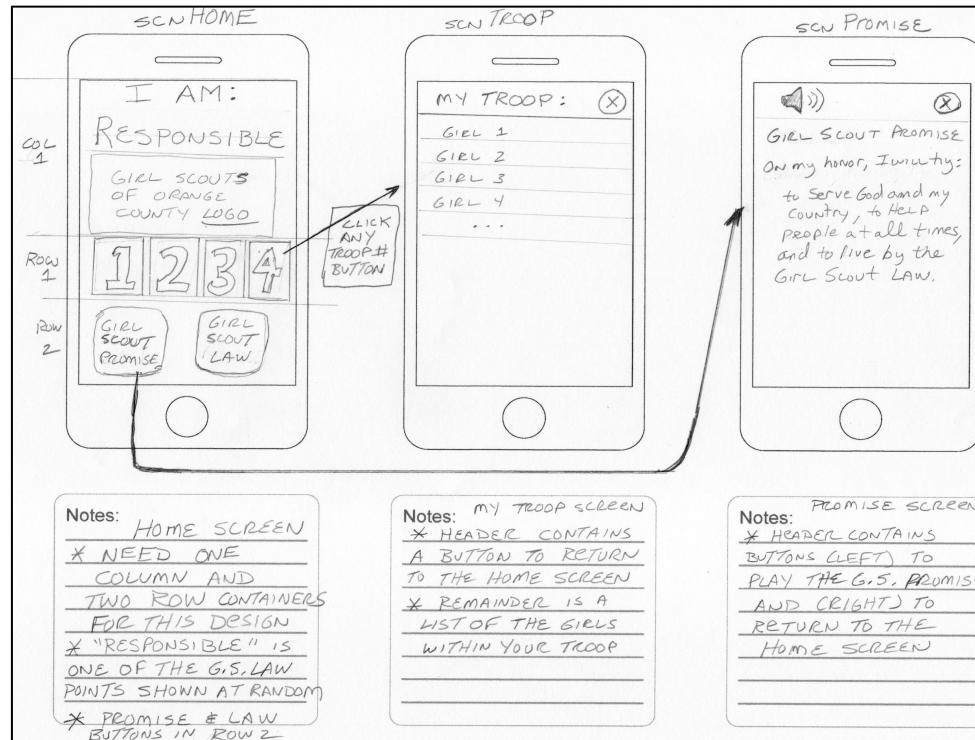
An app designer gives considerable thought to the user interface (UI) of the app. App development tools generally provide “containers” that hold UI components like buttons and sliders. The container determines the **alignment and orientation** of the components placed within it. Let’s drag three buttons onto our home screen in order to explore how layout works!



The screenshot shows the Thunkable app editor interface. On the left, the sidebar lists "Design" and "Blocks", the project name "MyGSOCAPP", and the scene "scnHome". Inside "scnHome", there are three buttons labeled "Button1", "Button2", and "Button3". Below them is a placeholder for "Invisible Components". The "User Interface" section includes components like "Button", "Label", "Text Input", "List Viewer", "Web Viewer", and "Switch". In the center, a smartphone-shaped canvas displays the three buttons stacked vertically. To the right, the "scnHome" properties panel is open, showing settings for "BackgroundColor" (set to `rgba(0, 174, 88, 1)`), "Vertical Alignment" (set to "center"), and "Horizontal Alignment" (set to "center"). A callout box on the right says "Let's discuss these options!" with arrows pointing to the alignment settings.

# Help! Take a step back - we need a plan!

App designers use **Paper Prototyping** to plan their app. An example is shown below. Here we've captured the functionality of the design, but not the final look (e.g. colors, graphic images, etc.), which we might capture in a second version of our paper prototype. Now create your own paper prototype!



# YOUR paper prototype - goals for our app

---

## Our Goals:

- Our app allows us to read (and listen to) the Girl Scout Law and Promise.
- Each time we open our app, we're reminded of one of the points of the Girl Scout Law (e.g. "Courageous").
- Our app displays the current members of our troop.

We will need a Home Screen, Troop Screen, Law Screen and Promise Screen to accomplish the goals above! [e.g. scnHome, scnTroop, scnLaw, scnPromise]

Our Home Screen should display random Girl Scout Law points, contain our troop number (clicking on our troop number navigates to the Troop Screen), as well as buttons for navigating to the Girl Scout Law and Promise screens.

***The rest of YOUR app design is up to you!!***

# YOUR design - our UI Assets



asterisk.png



asterisk2.png



Banana.png



btnGSLaw.jpg



btnGSPromise.jpg



cat1.png



cat4.png



clock.png



explosion.png



GirlScoutsMakeTheWorldA...ace.png



GS\_GIRL\_logo.jpg



gsappicon.gif



GSLaw.html



gslogofaces.jpg GSOC\_ServiceMark.jpg



GSPromise.html



icon\_back\_arrow.png



icon\_close\_clearbg.png



icon\_close\_w\_circleg.png



icon\_close\_x.png



icon\_face.png



icon\_happy.png



icon\_listening.png



icon\_phone\_talkingg.png



icon\_speaker.png



icon\_talk\_16px.png



icon\_talk\_30px.png



icon\_thumbs\_up.png



icon\_wrench.png JRTroopNbr4.jpg



lightningbolt.png



rubber-duck.png



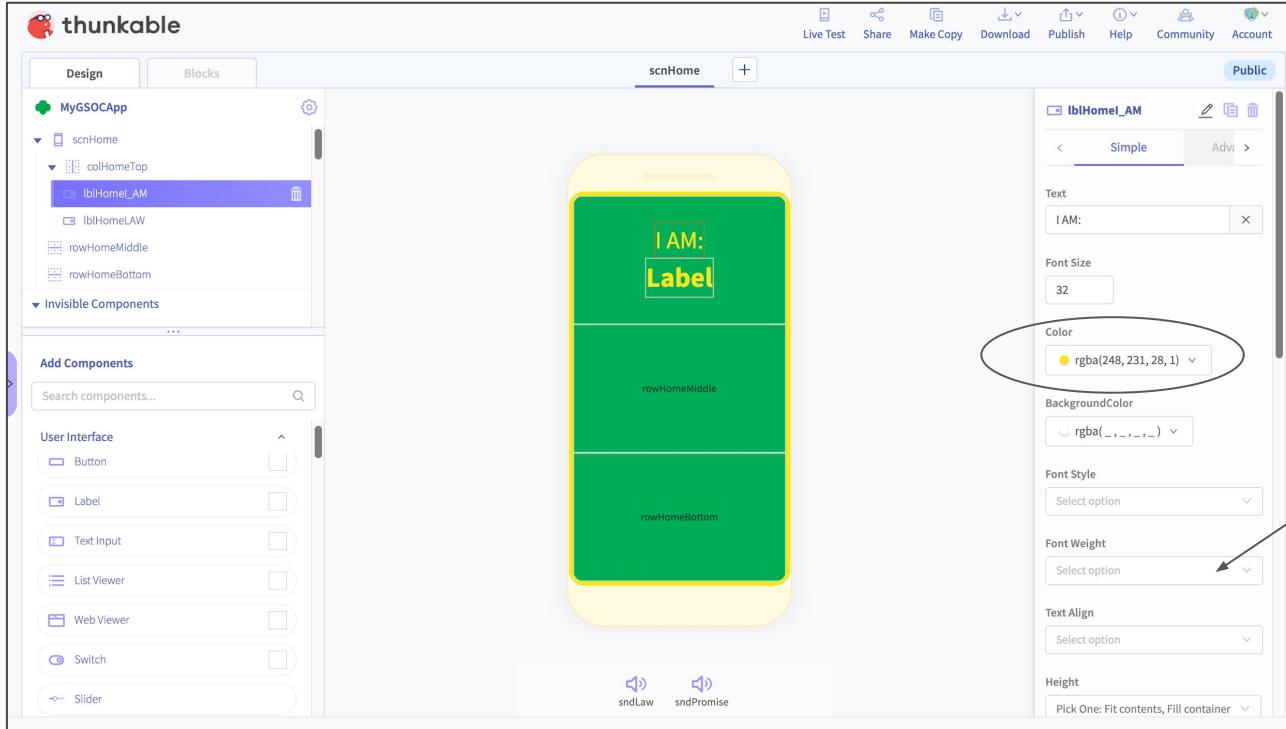
taco.png



unicorn.png

# Guided Design - scnHome

Add a column container and two row contains to the Home screen, and then add two Label components to the column container. Set the font sizes of the labels to 32, and use the same “yellow” color for the labels as our border. Bold the bottom label (lblHomeLAW).

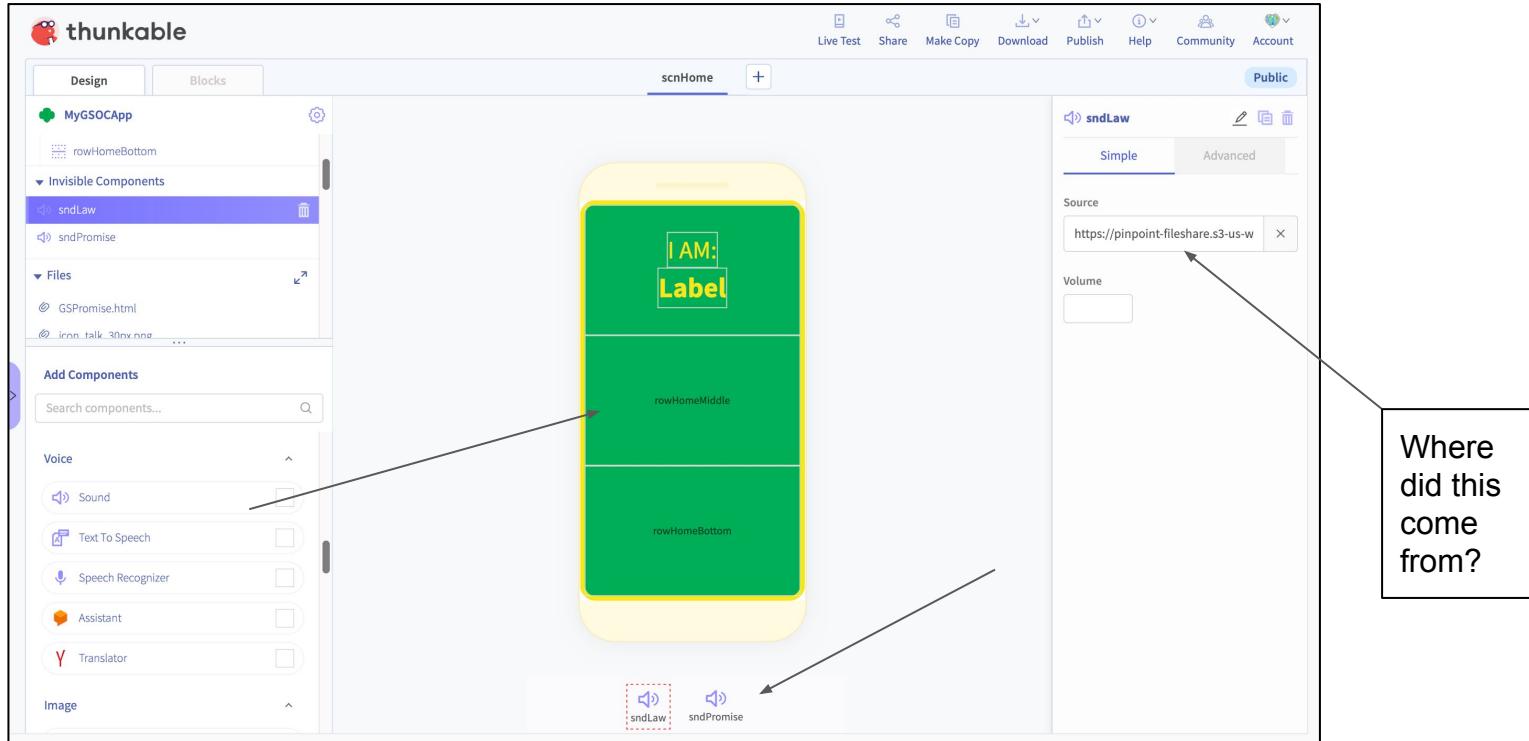


The screenshot shows the Thunkable app interface with the following details:

- Project Structure:** MyGSOCApp > scnHome > colHomeTop > lblHomeLAM (selected), lblHomeLAW.
- Design View:** A smartphone-shaped canvas displays three stacked components: "rowHomeMiddle" and "rowHomeBottom" at the bottom, and "lblHomeLAM" above them. The "lblHomeLAM" label contains the text "I AM: Label".
- Component Properties:** The "lblHomeLAM" component's properties panel is open:
  - Text:** I AM: Label
  - Font Size:** 32
  - Color:** rgba(248, 231, 28, 1) (yellow)
  - Background Color:** rgba(255, 255, 255, 1) (white)
  - Font Style:** Select option
  - Font Weight:** Select option (highlighted with a callout)
  - Text Align:** Select option
  - Height:** Pick One: Fit contents, Fill container
- Callout:** A callout box with an arrow points from the text "Use this parameter to set a bold font weight." to the "Font Weight" dropdown menu.

# Guided Design - Sound Components

Drag two Sound components onto scnHome. You'll note that the sound components display BELOW scnHome; these are “invisible components” as they do not appear within our app’s UI. Use the ATTRIBUTE PANE to name the sound components sndLaw and sndPromise.



thunkable

Design    Blocks    scnHome +

MyGSOCApp

Invisible Components

- sndLaw
- sndPromise

Files

- GSPromise.html
- iron\_talk\_30nxnng.mp3

Add Components

Search components...

- Voice
  - Sound
  - Text To Speech
  - Speech Recognizer
  - Assistant
  - Translator
- Image

rowHomeBottom

I AM:  
Label

rowHomeMiddle

rowHomeBottom

sndLaw    sndPromise

sndLaw

Simple    Advanced

Source

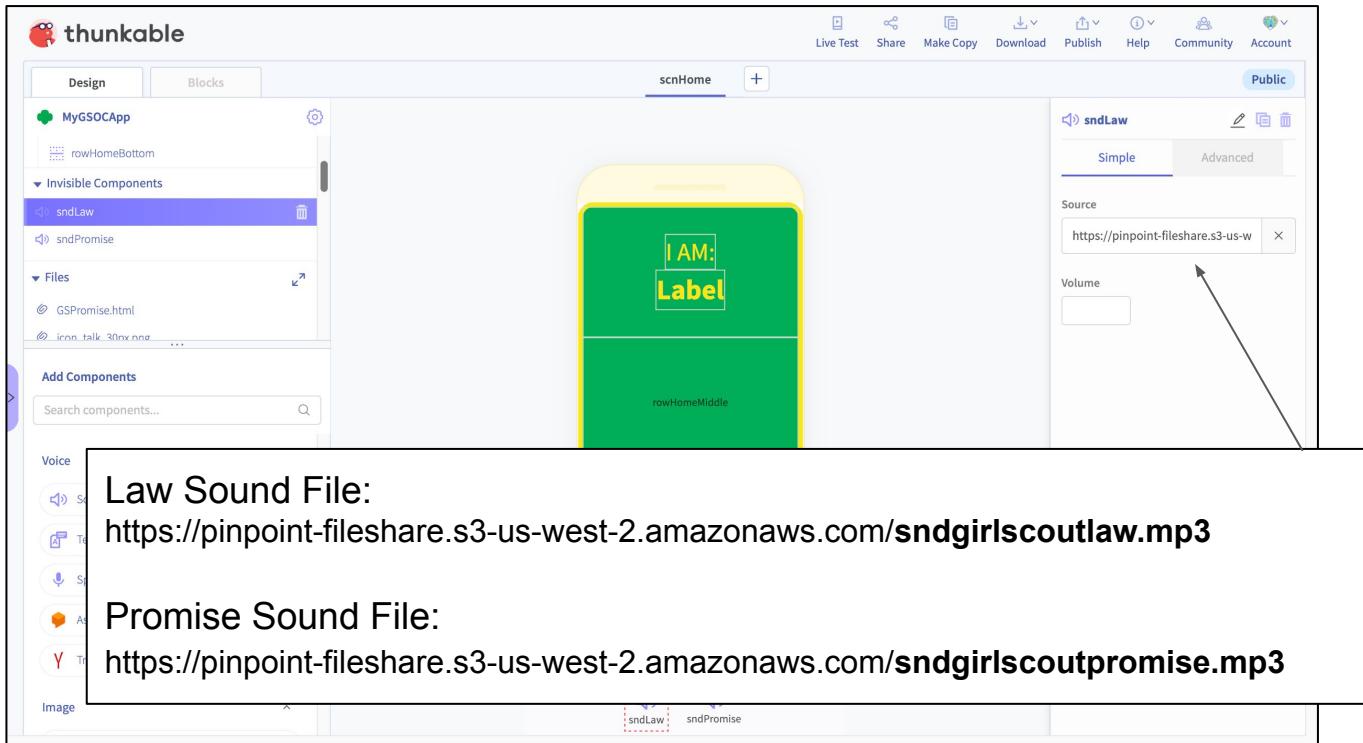
https://pinpoint-fileshare.s3-us-west-2.amazonaws.com/iron\_talk\_30nxnng.mp3

Volume

Where did this come from?

# Guided Design - Sound Components Source

Open the assets file CopyAndPasteLinks.txt using Notepad and **cut and paste the URL** for the LAW sound (.MP3) file into the sndLaw's ATTRIBUTE PANE **Source** parameter. Do the same for the sndPromise, using the PROMISE sound (.MP3) file URL.

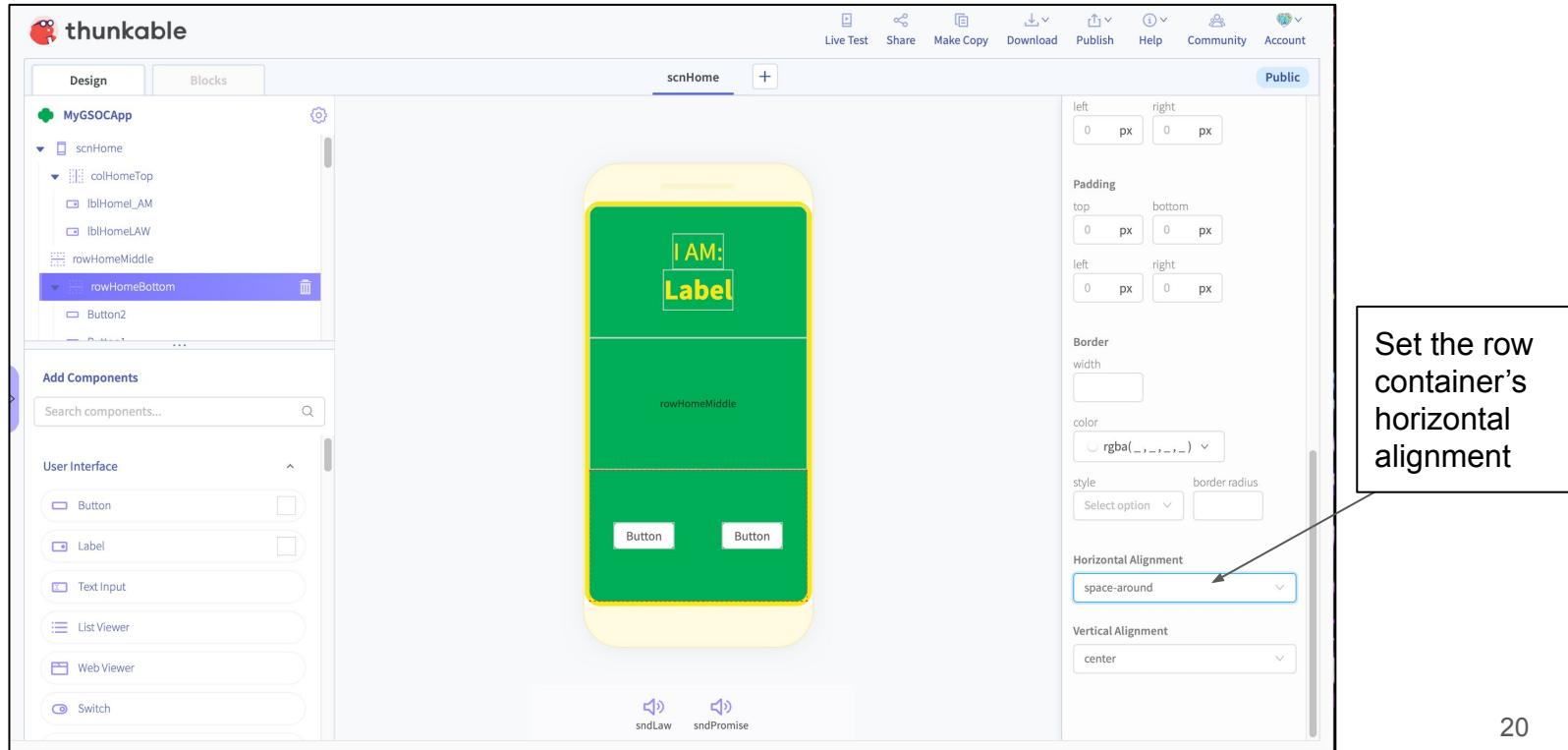


Law Sound File:  
<https://pinpoint-fileshare.s3-us-west-2.amazonaws.com/sndgirlscoutlaw.mp3>

Promise Sound File:  
<https://pinpoint-fileshare.s3-us-west-2.amazonaws.com/sndgirlscoutpromise.mp3>

# Guided Design - Law and Promise Buttons

Drag two buttons onto the rowHomeBottom container. Click on the container (outside of the buttons) to select it, then set the Horizontal Alignment in the ATTRIBUTE PANE of the row container to space-around.



Thunkable

Design Blocks scnHome + Public

MyGSOCApp

scnHome

colHomeTop

lblHomeLAM

lblHomeLAW

rowHomeMiddle

rowHomeBottom

Button2

Add Components

Search components...

User Interface

Button

Label

Text Input

List Viewer

Web Viewer

Switch

Live Test Share Make Copy Download Publish Help Community Account

left right  
0 px 0 px

Padding

top bottom  
0 px 0 px

left right  
0 px 0 px

Border

width

color

style border radius

Horizontal Alignment

space-around

Vertical Alignment

center

I AM:  
Label

rowHomeMiddle

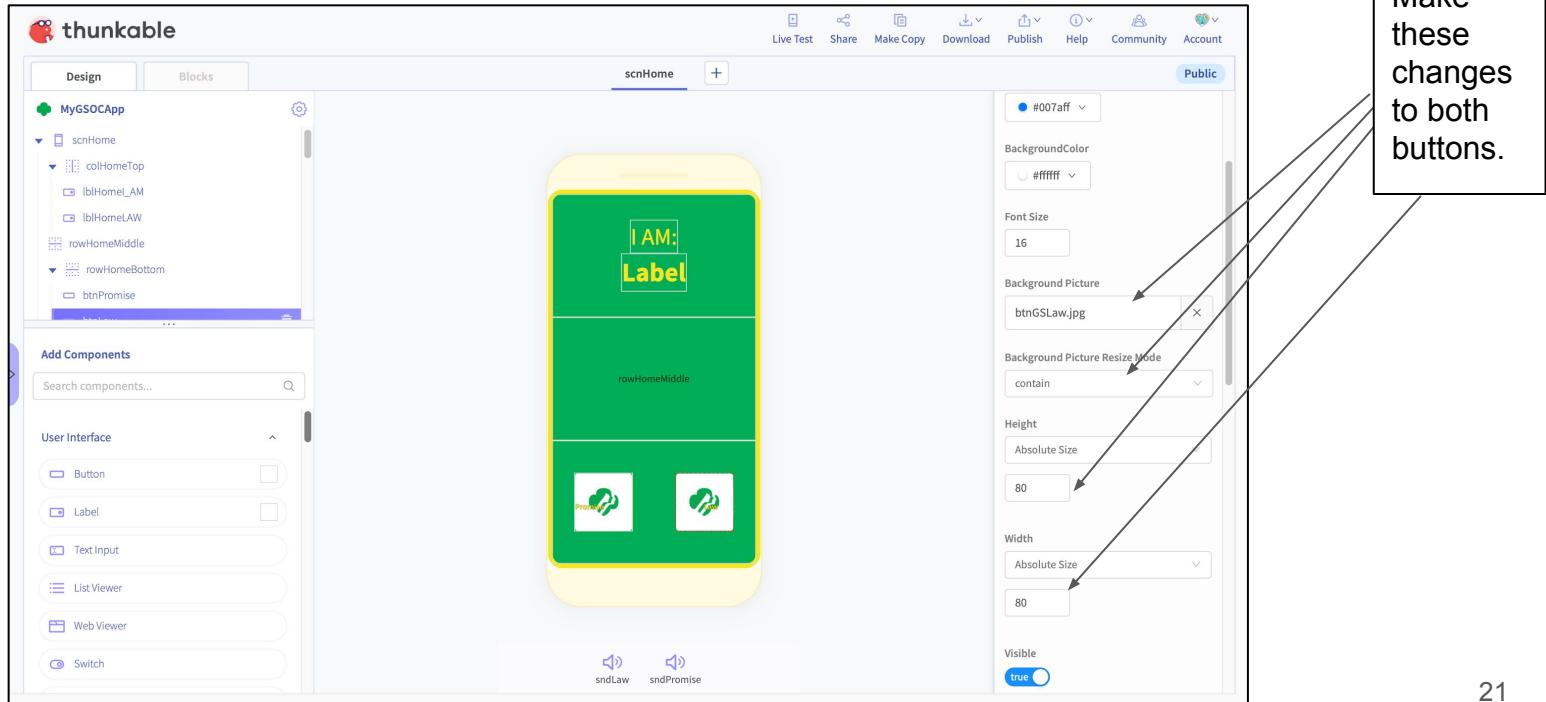
Button Button

sndLaw sndPromise

Set the row container's horizontal alignment

# Guided Design - Law and Promise Buttons

Select each button in order, changing the name (e.g. btnLaw) and selecting a Background Picture for the button face. Resize the button so the picture fits better, and change the background resize mode to “contain”. We used a value of 80 for the height and width of the button, but you may wish to adjust that after viewing your app on your phone.

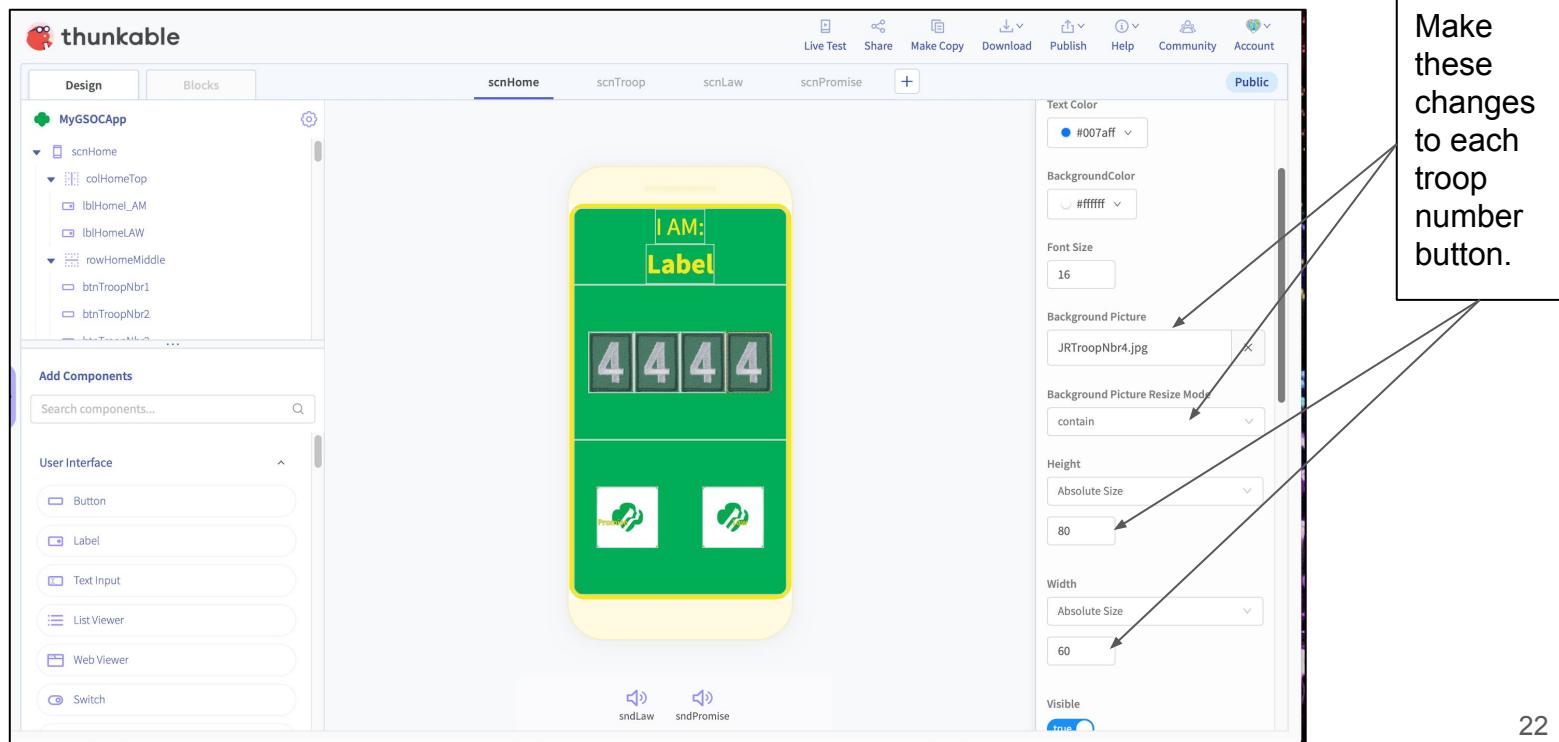


The screenshot shows the Thunkable app editor interface. On the left, the project tree shows 'MyGSOCApp' with 'scnHome' selected, containing 'colHomeTop', 'rowHomeMiddle', and 'rowHomeBottom' sections, along with a 'btnPromise' component. The main canvas displays a smartphone screen with a green background. Inside the screen, there is a large green button with the text 'I AM: Label' and two smaller green buttons at the bottom. The right side of the interface is the properties panel for the 'btnPromise' component. It includes fields for 'BackgroundColor' (blue hex code), 'Font Size' (16), 'Background Picture' (set to 'btnGSLaw.jpg'), 'Background Picture Resize Mode' (set to 'contain'), and 'Height' and 'Width' both set to '80'. A callout box with the text 'Make these changes to both buttons.' has arrows pointing to the 'Background Picture', 'Background Picture Resize Mode', 'Height', and 'Width' fields.

Make these changes to both buttons.

# Guided Design - Troop Number Buttons

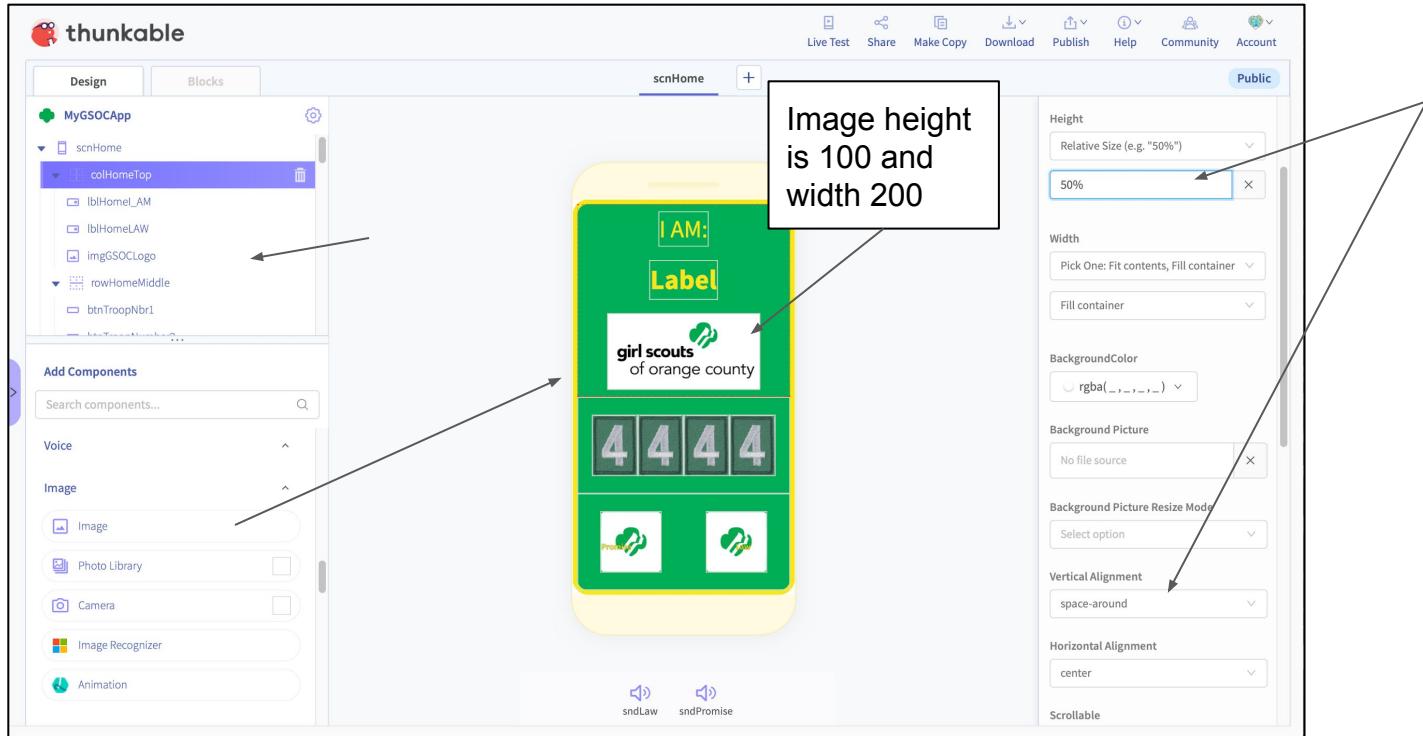
How many digits in your troop number? Add that many buttons into the middle row container, then set the background picture for each button appropriately. Set the Background resize mode to contain, height to 80 and width to 60. Did you name your buttons btnTroopNbr1 (etc.)?



Make these changes to each troop number button.

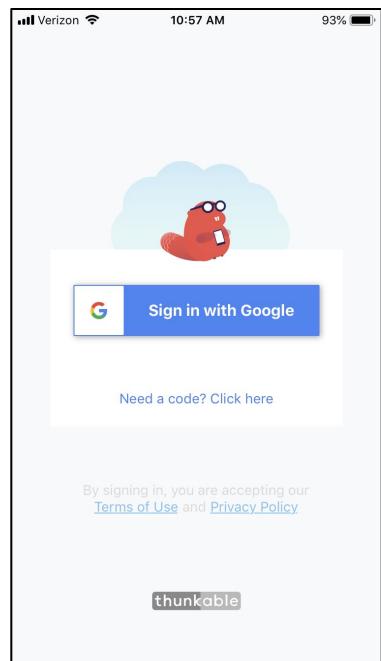
# Guided Design - Add a logo!

Drag an Image component onto the column container, being sure to place it beneath our lblHomeLaw label. **Wow, too tight!** Click on the column container and set its height to 50% and its vertical alignment to space-around. Add a picture to the image component and rename it!



# See how it looks (so far) on our phone!

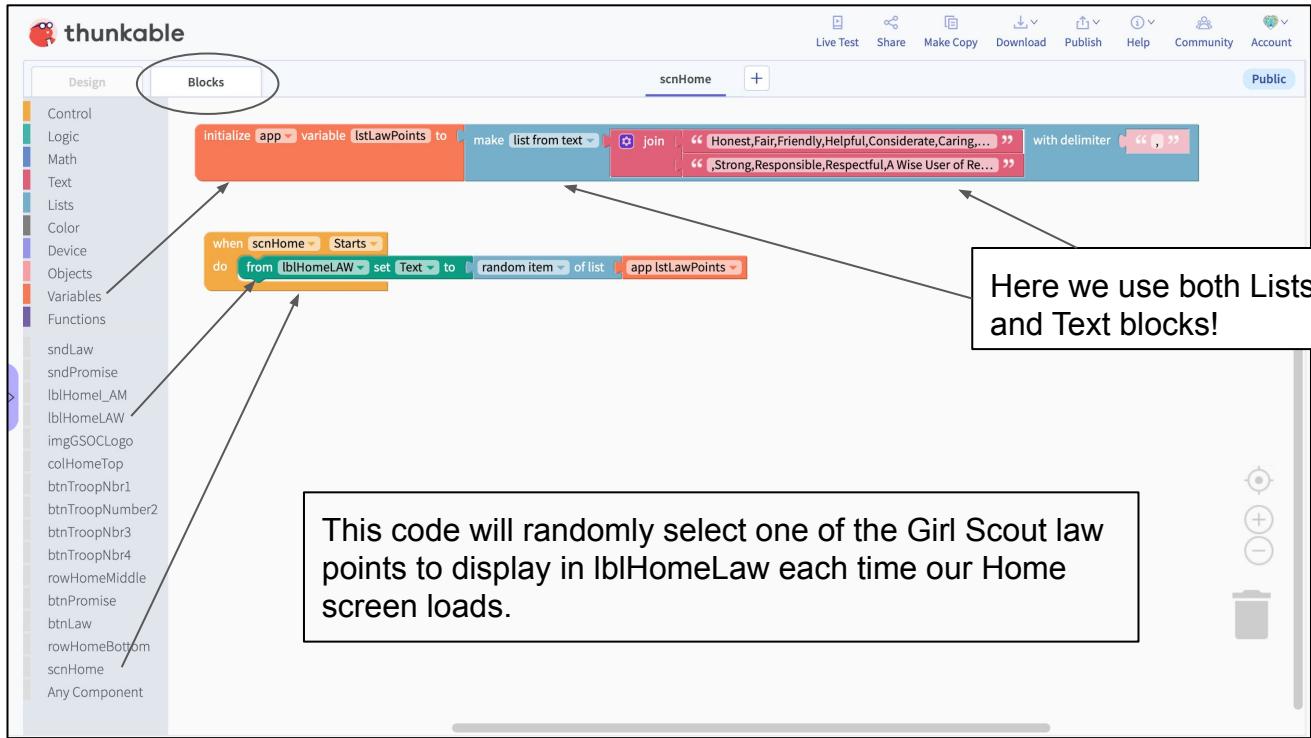
After installing the Thunkable Live app (App Store or Google Play), log in using your GMAIL ACCOUNT, then take a peek at your app screen, it's coming along nicely!



Wait! How did  
THIS get on  
there??

# Guided Design - Let's Code!

Initialize a variable to a list which we'll build from text that we copy from our assets file CopyAndPasteLinks.txt. Click on scnHome and add a Starts block, setting our lblHomeLaw to a random item within our newly created list variable.

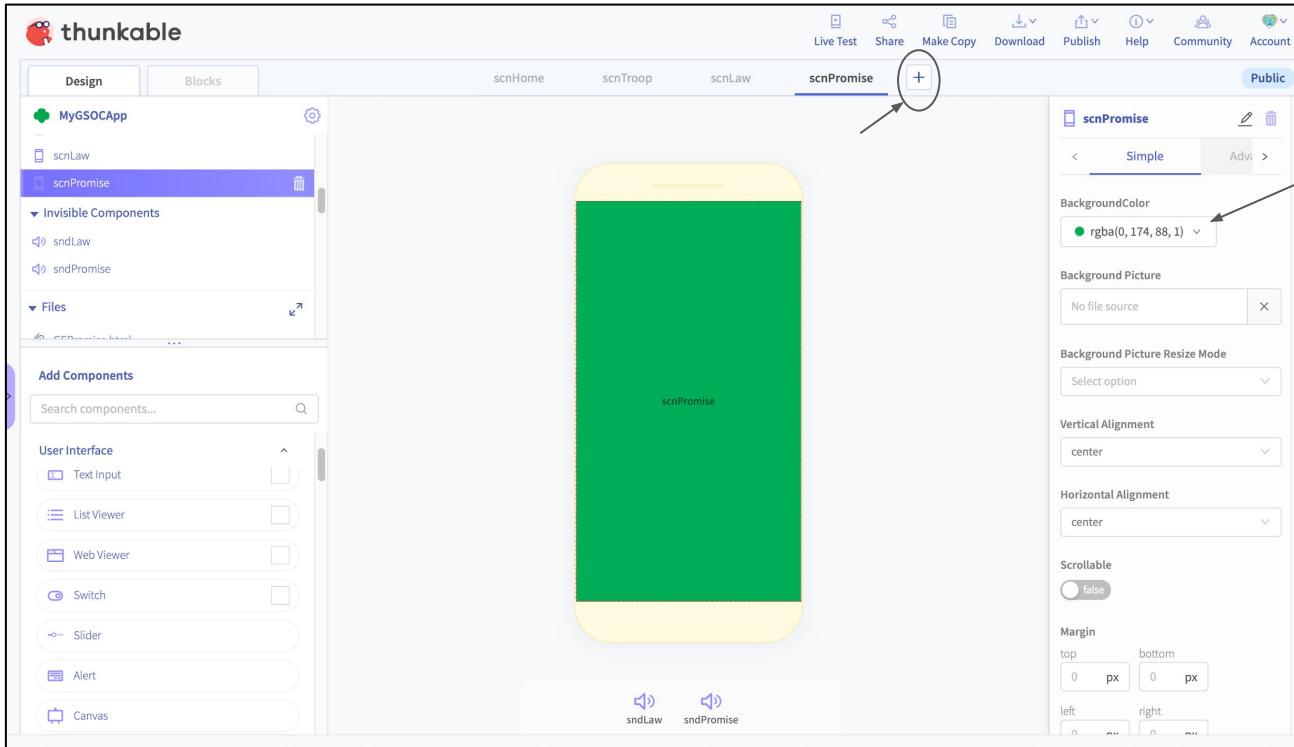


The screenshot shows the thunkable app interface with the following details:

- Blocks Tab:** The "Blocks" tab is selected in the top navigation bar.
- Variables:** A variable named "lstLawPoints" is defined with the value: "Honest,Fair,Friendly,Helpful,Considerate,Caring,...", "Strong,Responsible,Respectful,A Wise User of Re...", and a delimiter of ",".
- Scenes:** The scene is titled "scnHome".
- Script:** A script attached to "scnHome" starts with a "when [scnHome] starts" hat block, followed by a "do" loop containing a "set [lblHomeLaw] to [random item of lstLawPoints]" block.
- Scratch-like Blocks:** The "Blocks" palette on the left lists various categories like Control, Logic, Text, Lists, etc., with specific blocks for "sndLaw", "sndPromise", "lblHomeLAW", "imgGSOCLogo", "colHomeTop", "btnTroopNbr1", "btnTroopNumber2", "btnTroopNbr3", "btnTroopNbr4", "rowHomeMiddle", "btnPromise", "btnLaw", "rowHomeBottom", "scnHome", and "Any Component".
- Annotations:**
  - A callout box points to the "Lists" and "Text" blocks in the script: "Here we use both Lists and Text blocks!"
  - A callout box points to the "random item of list" block: "This code will randomly select one of the Girl Scout law points to display in lblHomeLaw each time our Home screen loads."

# Guided Design - adding screens!

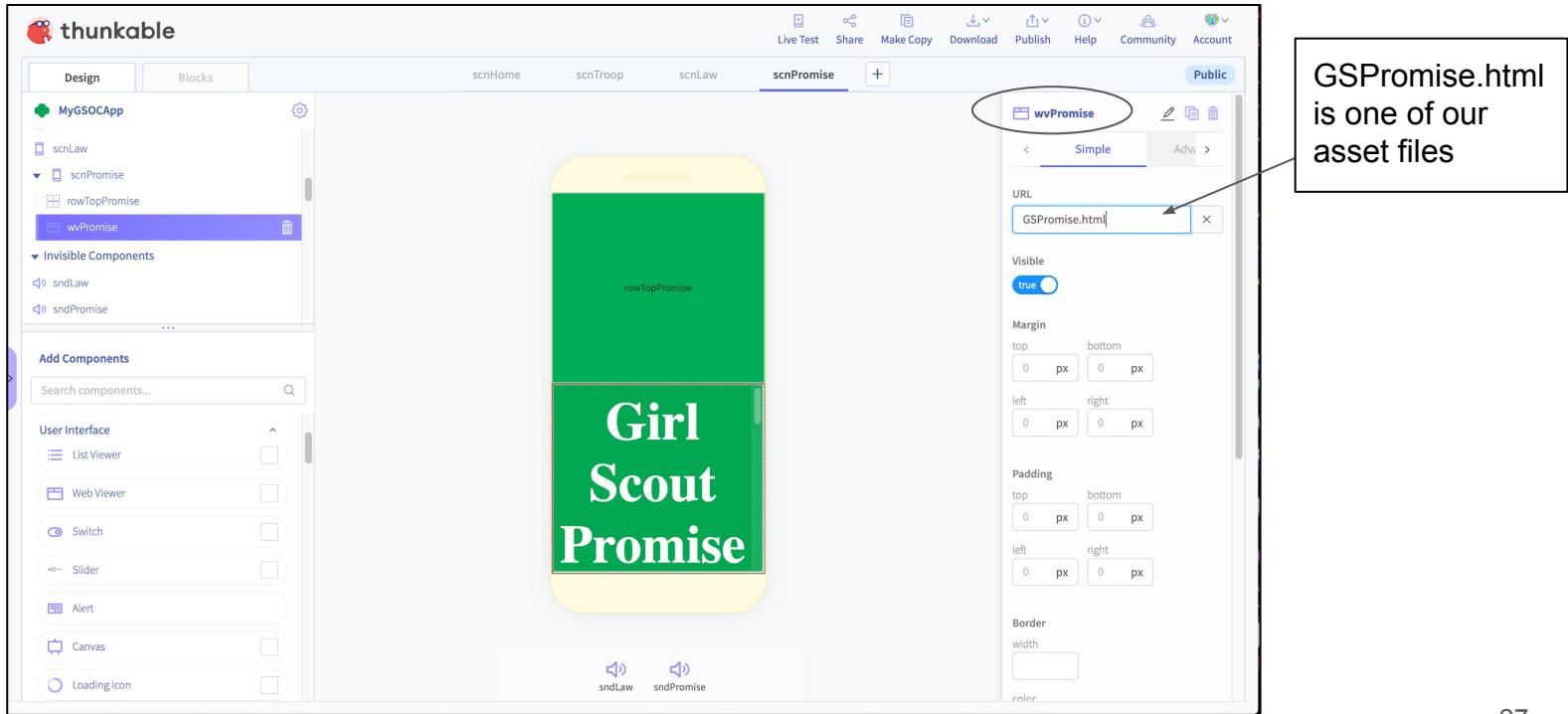
Use the + button to the right of the screen names to add **three** new screens; scnTroop, scnLaw and scnPromise. Change the background color on each of the three screens to the Girl Scout Colors, R = 0, G = 174 and B = 88 as we did with scnHome.



The screenshot shows the thunkable app interface. On the left, there's a sidebar with 'Design' and 'Blocks' tabs, a project tree ('MyGSOCApp' with 'scnLaw' and 'scnPromise'), and component categories like 'User Interface' (Text Input, List Viewer, Web Viewer, Switch, Slider, Alert, Canvas). In the center, a smartphone-shaped canvas displays the 'scnPromise' screen, which has a solid green background. At the top of the screen, there are buttons for 'Live Test', 'Share', 'Make Copy', 'Download', 'Publish', 'Help', 'Community', and 'Account'. Below the screen, there are two sound wave icons labeled 'sndLaw' and 'sndPromise'. On the right, the 'scnPromise' screen card shows its properties: 'BackgroundColor' is set to 'rgba(0, 174, 88, 1)', 'Background Picture' is 'No file source', 'Background Picture Resize Mode' is 'Select option', 'Vertical Alignment' is 'center', 'Horizontal Alignment' is 'center', 'Scrollable' is 'false', and 'Margin' is 'top: 0px, bottom: 0px, left: 0px, right: 0px'. A blue circle highlights the '+' button to the right of 'scnPromise' in the top navigation bar, and a red arrow points from this circle to the 'BackgroundColor' dropdown in the properties panel.

# Guided Design - Web Viewer Component

Drag a row container and a Web Viewer component onto the Promise screen. Type in the asset filename “GSPromise.html” into the web viewer component’s URL parameter on the ATTRIBUTE PANE of the web viewer. Note that we renamed the row and web viewer components.

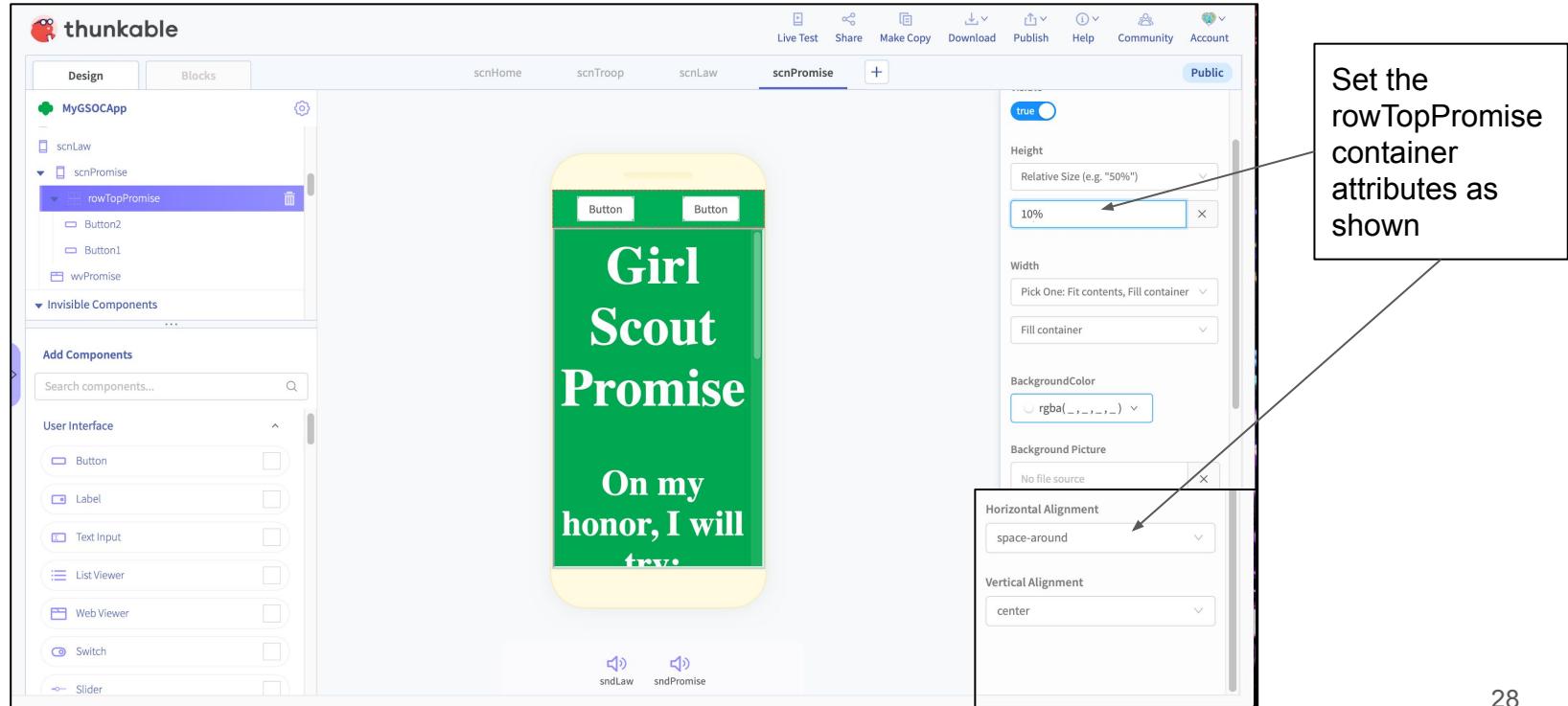


The screenshot shows the Thunkable app interface for mobile application development. On the left, the project tree for "MyGSOCApp" shows a "scnPromise" scene with a "rowTopPromise" row and a "wvPromise" Web Viewer component. The main canvas displays a smartphone screen with a green background and the text "rowTopPromise" at the top and "Girl Scout Promise" in large white letters below. The "Blocks" tab is selected in the top navigation bar. On the right, the "Attribute pane" for the "wvPromise" component is open, showing the "URL" field set to "GSPromise.html". A callout box points from the text "GSPromise.html is one of our asset files" to this URL field. The attribute pane also includes sections for "Public", "Visible" (set to true), "Margin", "Padding", and "Border".

GSPromise.html  
is one of our  
asset files

# Guided Design - Promise Screen UI

Resize the row container so it represents only 10% of the height of our screen. Set the horizontal alignment of the row container to “space-around”, then add two buttons to the row container. We’ll use these buttons to play our promise sound file and to close the promise screen.



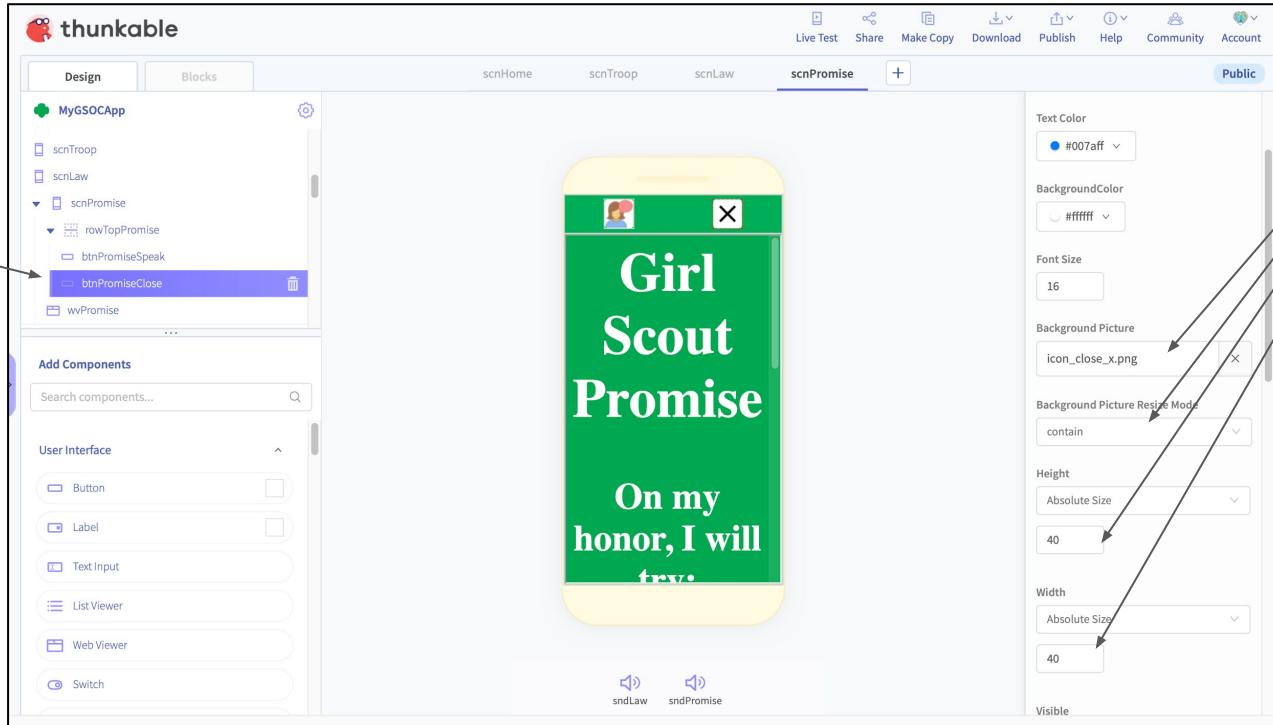
The screenshot shows the thunkable app interface with the following details:

- Left Sidebar:** Shows the project structure under "MyGSOCApp". The "scnPromise" scene is selected. Inside "scnPromise", there is a "rowTopPromise" component which contains "Button2" and "Button1". Below this is a "wvPromise" component and an "Invisible Components" section.
- Middle View:** Displays a preview of the "scnPromise" scene. It features a green background with white text: "Girl Scout Promise" at the top and "On my honor, I will trv." below it. At the bottom are two small speaker icons labeled "sndLaw" and "sndPromise".
- Right Panel (Properties):** Shows the properties for the "rowTopPromise" component.
  - Height:** Set to "10%" (highlighted with a red arrow).
  - Width:** Set to "Fit contents, Fill container".
  - Background Color:** Set to "rgba(0, 128, 0)".
  - Horizontal Alignment:** Set to "space-around".
  - Vertical Alignment:** Set to "center".
- Callout Box:** A callout box on the right side provides instructions: "Set the rowTopPromise container attributes as shown". It points to the "Height" and "Horizontal Alignment" settings in the properties panel.

# Guided Design - Promise Screen UI

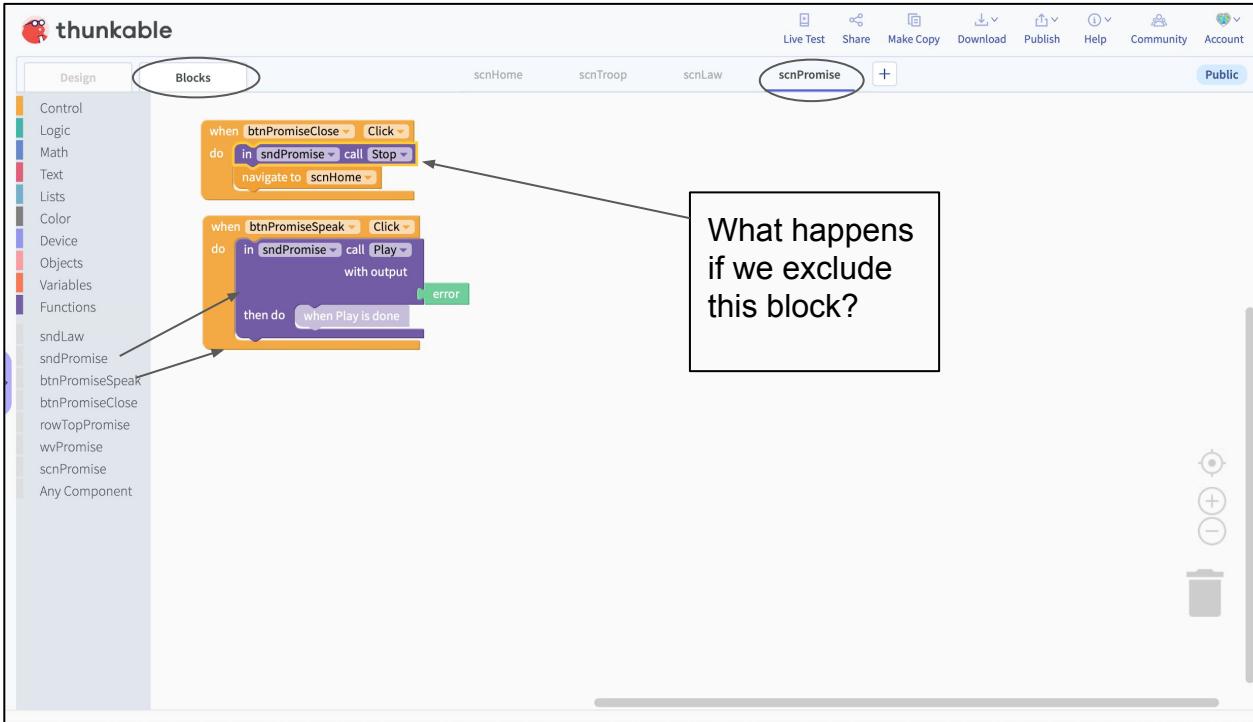
Name the two buttons based upon our conventions, and then add background pictures. Be sure to set the Background Picture parameter of each button to “contain”. A height and width of 40 looks good on the phone for the background pictures we have selected in this example.

Be sure to  
rename your  
buttons!



# Guided Design - Let's Code! (scnPromise)

Click on Blocks, then add “Click” event blocks for our Promise screen’s Speak and Close buttons. When we click the Close button, we return to the home screen. When we click the Speak button, we play our Promise sound file asset.

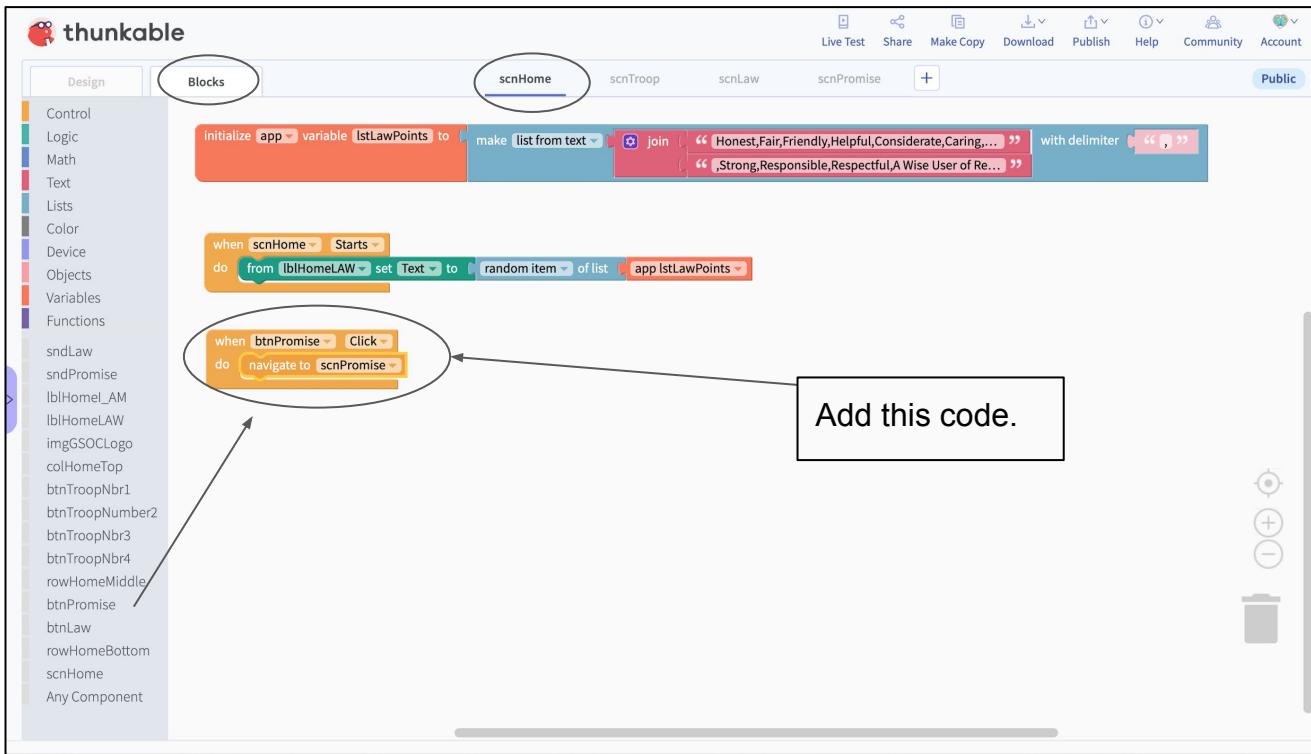


The screenshot shows the Thunkable app interface with the following details:

- Design Tab:** The active tab, indicated by a blue border.
- Blocks Tab:** The tab where the code blocks are located.
- Scenes:** A navigation bar with tabs: scnHome, scnTroop, scnLaw, and **scnPromise** (highlighted with a blue oval).
- Toolbar:** Standard toolbar with icons for Live Test, Share, Make Copy, Download, Publish, Help, Community, and Account.
- Code Area:** The workspace where blocks are placed.
  - Close Button Block:** An orange "when btnPromiseClose Click" block with a "do" loop containing "in sndPromise call Stop" and "navigate to scnHome".
  - Speak Button Block:** An orange "when btnPromiseSpeak Click" block with a "do" loop containing "in sndPromise call Play with output" and a green "error" block. This block has a "then do" branch pointing to an orange "when Play is done" block.
- Object List:** A sidebar on the left listing objects: Control, Logic, Math, Text, Lists, Color, Device, Objects, Variables, Functions, and several specific components like sndLaw, sndPromise, btnPromiseSpeak, btnPromiseClose, etc.
- Question Box:** A callout box with the text "What happens if we exclude this block?" pointing to the "when Play is done" block.

# Guided Design - Let's Code! (scnHome)

But how do we reach our new Promise screen? Click back on scnHome and then Blocks and add a Click event block for our btnPromise. Now we can click on bthPromise and navigate to scnPromise.



The screenshot shows the Thunkable app editor with the scnHome scene selected. The sidebar on the left lists various components and variables. Two scripts are visible in the main workspace:

- Script 1 (Top):** Triggered by "when scnHome Starts". It sets the text of the component "lblHomeLaw" to a "random item" from a list variable named "lstLawPoints".
- Script 2 (Bottom):** Triggered by "when btnPromise Click". It contains the code "navigate to scnPromise".

A callout box with the text "Add this code." points to the "when btnPromise Click" block in the second script.

# Guided Design - Repeat for scnLaw

---

## For scnLaw:

Slide 26: Add the Girl Scout Background Color (R = 0, G =174, B = 88)

Slide 27: Add a row and web viewer component, and set the web viewer's URL parameter to the asset file **GSLaw.html**.

Slide 28: Change the row container's height to 10% of our screen space, and set the horizontal alignment of the row container to “space around”.

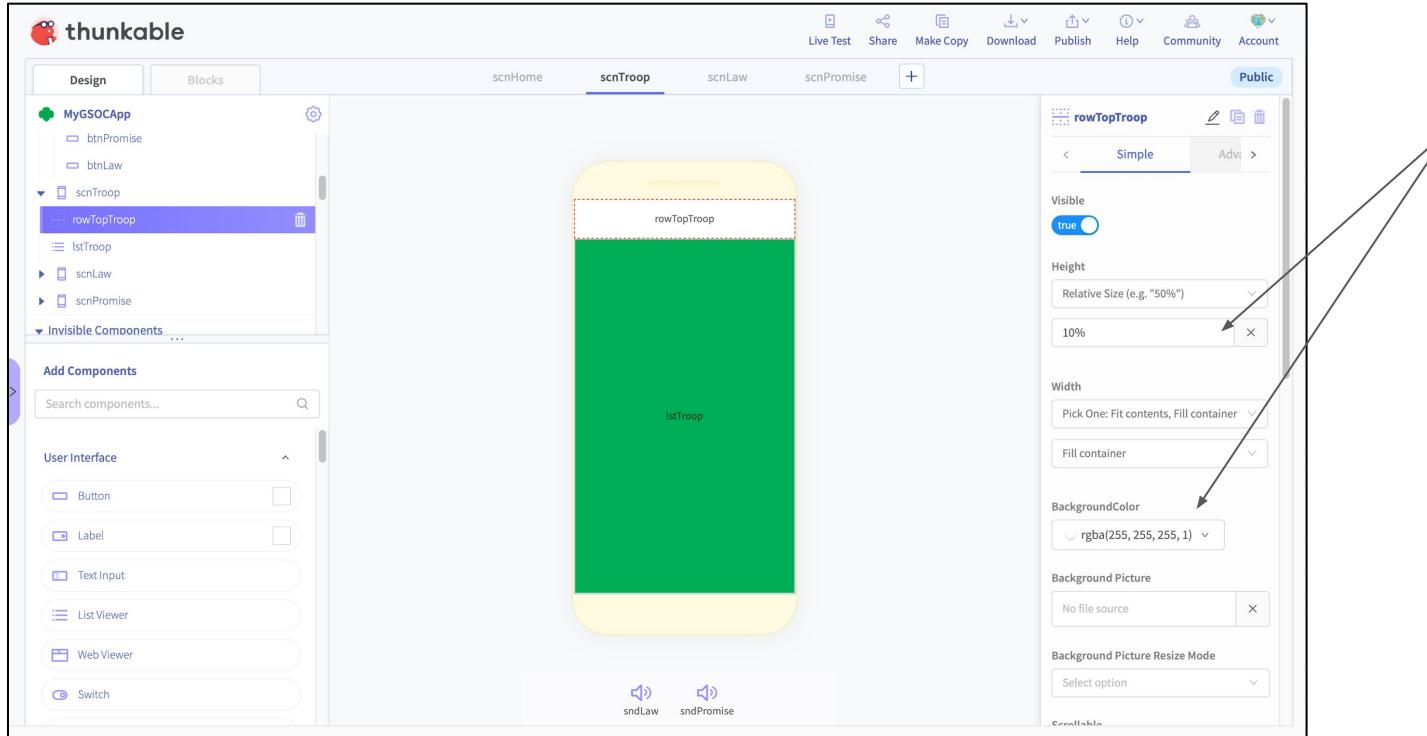
Slide 29: Add images to the row container's buttons and set their height / width to 40.

Slide 30: Add code to scnLaw, causing btnLawClose to navigate to scnHome and btnLawSound to play the sndLaw sound file asset.

Slide 31: Add code to scnHome, causing btnLaw to navigate to scnLaw.

# Guided Design - Your Troop (list)

Select scnTroop, then add a row container (top) and List Viewer component to scnTroop, renaming using our conventions. Set the row container's relative height to 10% and (to be different!) set the background color of the row container to white (R=255, G=255, B=255).

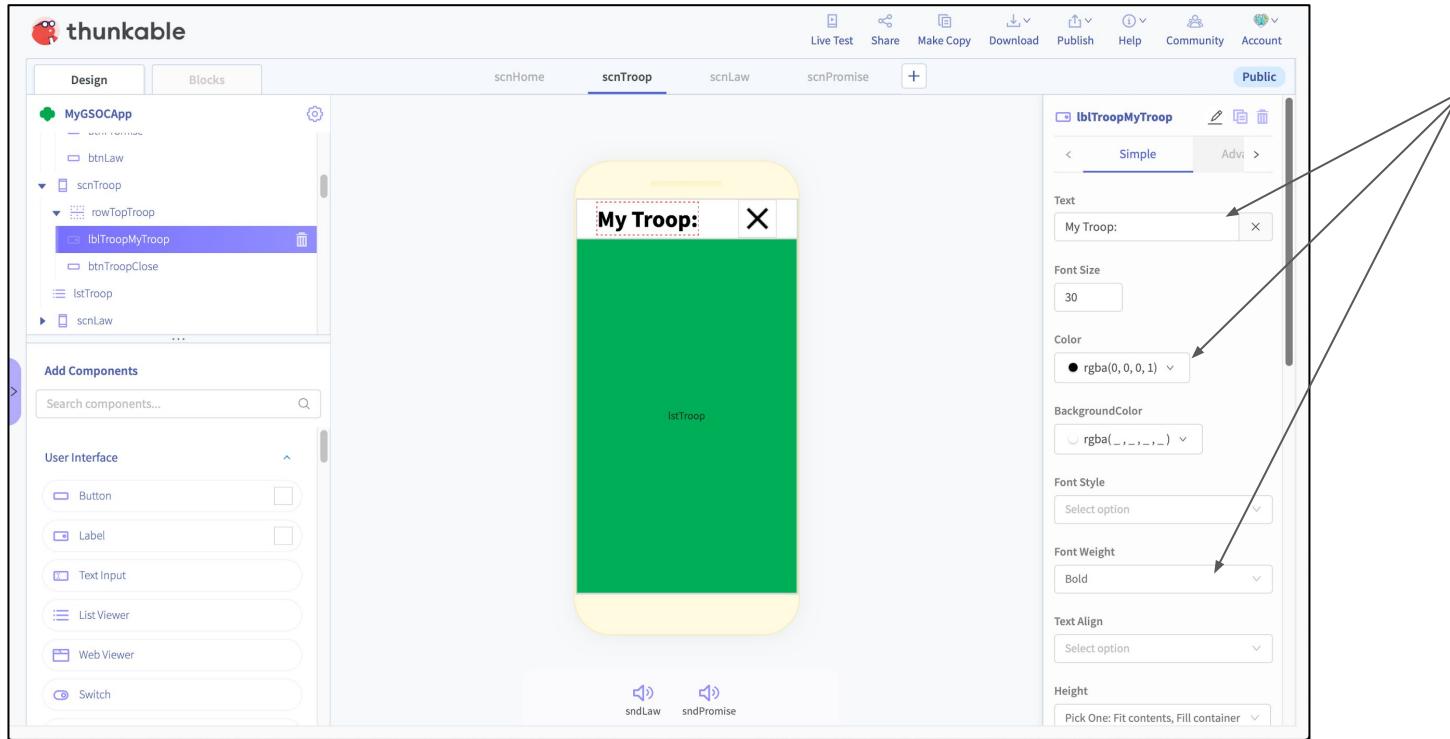


The screenshot shows the thunkable app interface with the following details:

- Project Structure:** The project is named "MyGSOCApp". It contains components: btnPromise, btnLaw, scnHome, scnTroop, scnLaw, and scnPromise. scnTroop is currently selected.
- Design View:** A smartphone screen is displayed with a yellow header bar labeled "rowTopTroop" and a green body labeled "lstTroop".
- Component Properties Panel:** The "rowTopTroop" component is selected for configuration. The "Height" section is set to "Relative Size (e.g. "50%")" with the value "10%" highlighted by a red box and an arrow pointing to it from the text above.
- Background Color:** The "BackgroundColor" dropdown is set to "rgba(255, 255, 255, 1)" (white).
- Sound Effects:** Two sound effect icons are present at the bottom: "sndLaw" and "sndPromise".

# Guided Design - Your Troop (list)

Add a label and a button to the rowTopTroop container. Change the text parameter within the ATTRIBUTES PANE to My Troop: with a font size of 30. Set the text color to black and the font weight to bold.

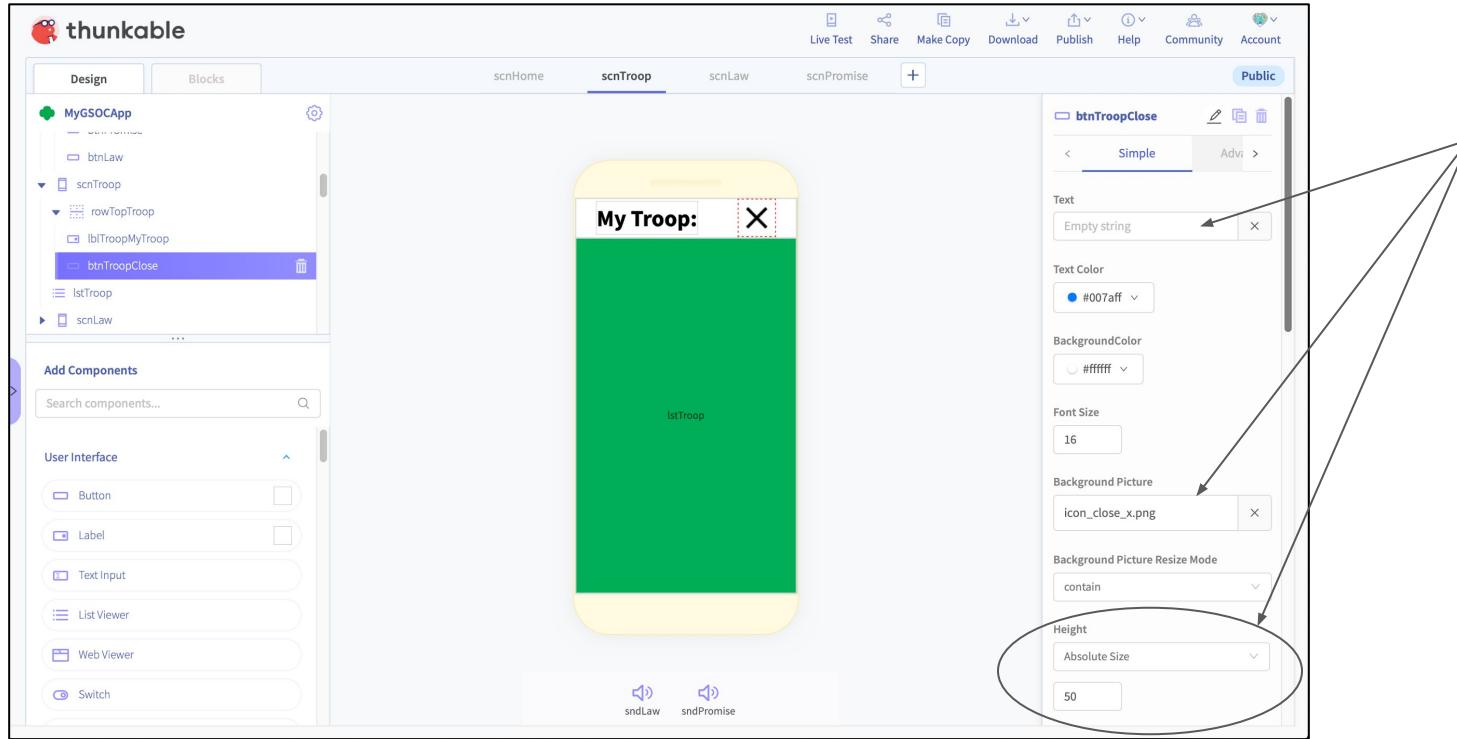


The screenshot shows the Thunkable app interface with the following details:

- Design View:** The main area displays a smartphone screen with the title "My Troop:" at the top. Below it, the text "lstTroop" is visible.
- Scenes:** The top navigation bar shows scenes: scnHome, scnTroop (selected), scnLaw, and scnPromise.
- Object Tree:** On the left, the object tree shows:
  - MyGSOCApp
    - btnLaw
    - scnTroop
      - rowTopTroop
        - lblTroopMyTroop
        - btnTroopClose
      - lstTroop
    - scnLaw
- Components Panel:** The "User Interface" section lists components: Button, Label, Text Input, List Viewer, Web Viewer, and Switch.
- Properties Panel (for lblTroopMyTroop):** This panel is open on the right, showing settings for the label component:
  - Text:** My Troop:
  - Font Size:** 30
  - Color:** `rgba(0, 0, 0, 1)`
  - Font Style:** Select option
  - Font Weight:** Bold
  - Text Align:** Select option
  - Height:** Pick One: Fit contents, Fill containerThree arrows point from the text "My Troop:", the font size "30", and the font weight "Bold" to their respective settings in the properties panel.

# Guided Design - Your Troop (list)

Browse an image from our assets as the Background Picture for our close button. Don't forget to remove the "Button" text within the Text parameter field! We set the button's height and width to 50 in this example.

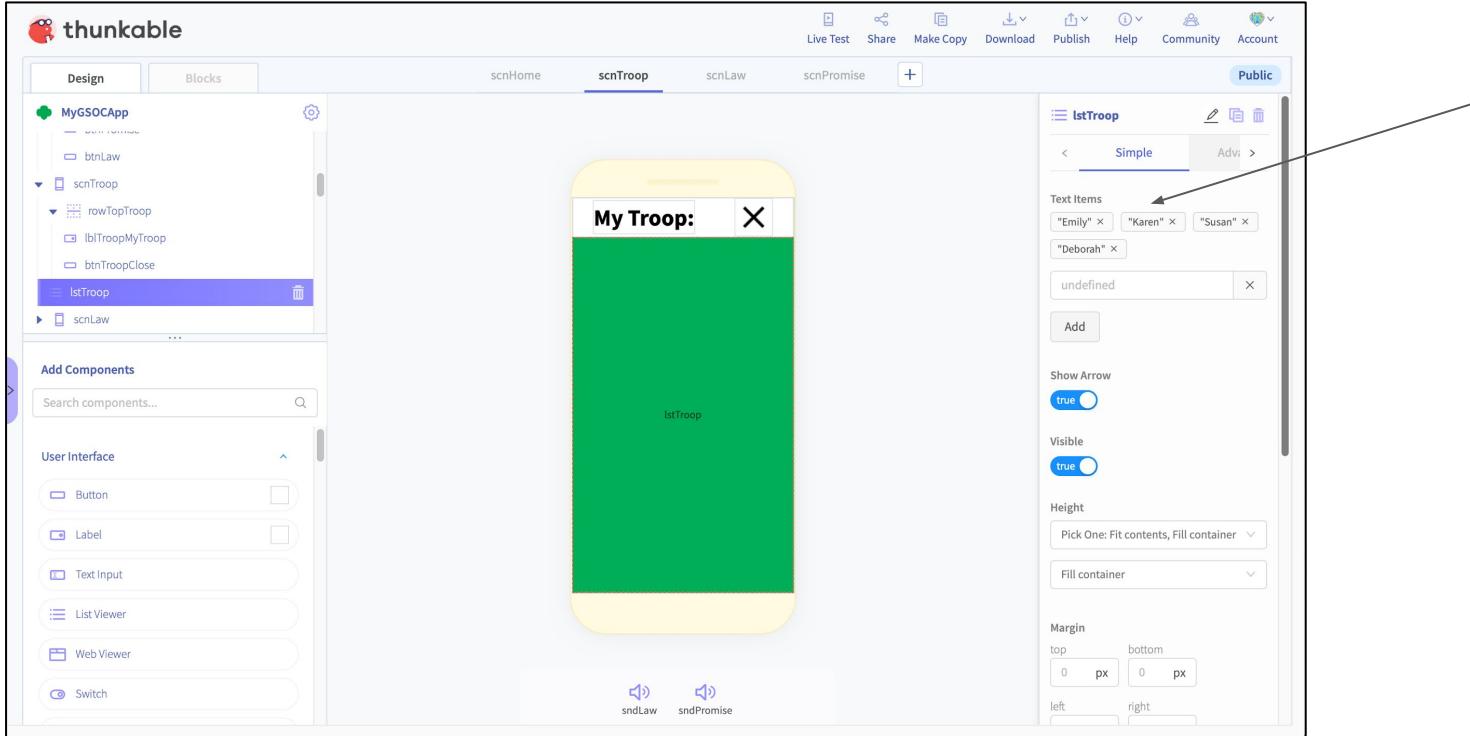


The screenshot shows the Thunkable app interface with the following details:

- Project Structure:** The project is named "MyGSOCApp". The current screen is "scnTroop". Other screens include "scnHome", "scnLaw", and "scnPromise".
- Component List:** The "My Troop" screen contains several components: "btnLaw", "scnTroop", "rowTopTroop", "lblTroopMyTroop", "btnTroopClose", "lstTroop", and "scnLaw".
- Add Components:** A sidebar lists components like "User Interface" (Button, Label, Text Input, List Viewer, Web Viewer, Switch), "Media" (Image, Video, Audio), and "Data" (List, Dictionary, JSON).
- Component Properties (btnTroopClose):**
  - Text:** Empty string
  - Text Color:** #007aff
  - Background Color:** #ffffff
  - Font Size:** 16
  - Background Picture:** icon\_close\_x.png
  - Background Picture Resize Mode:** contain
  - Height:** Absolute Size (set to 50)
- Screen Preview:** The screen shows a yellow header bar with the text "My Troop:" and a red close button. The main area is green and labeled "lstTroop".
- Bottom Buttons:** "sndLaw" and "sndPromise" are visible at the bottom of the screen.

# Guided Design - Your Troop (list)

Select the List Viewer component and then type in the names of the girls within your troop as the Text Items parameter.

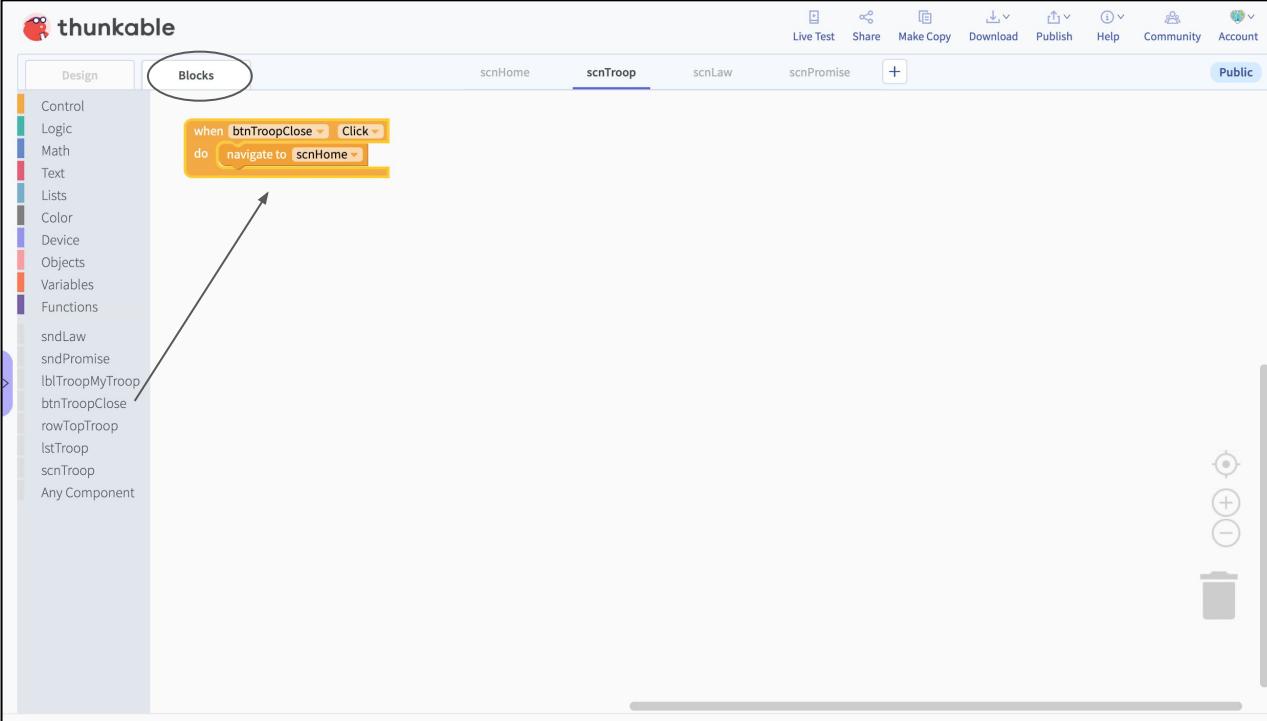


The screenshot shows the thunkable app interface with the following details:

- Project Structure:** MyGSOCApp > scnTroop > lstTroop
- Screen Preview:** A smartphone screen titled "My Troop:" displays a green background with the text "lstTroop".
- List Viewer Component Settings (for lstTroop):**
  - Text Items:** "Emily" X, "Karen" X, "Susan" X, "Deborah" X
  - undefined** (input field)
  - Add** (button)
  - Show Arrow:** true (radio button)
  - Visible:** true (radio button)
  - Height:** Pick One: Fit contents, Fill container (dropdown)
    - Fit contents
    - Fill container
  - Margin:** top 0 px, bottom 0 px, left right (input fields)

# Guided Design - Let's Code!

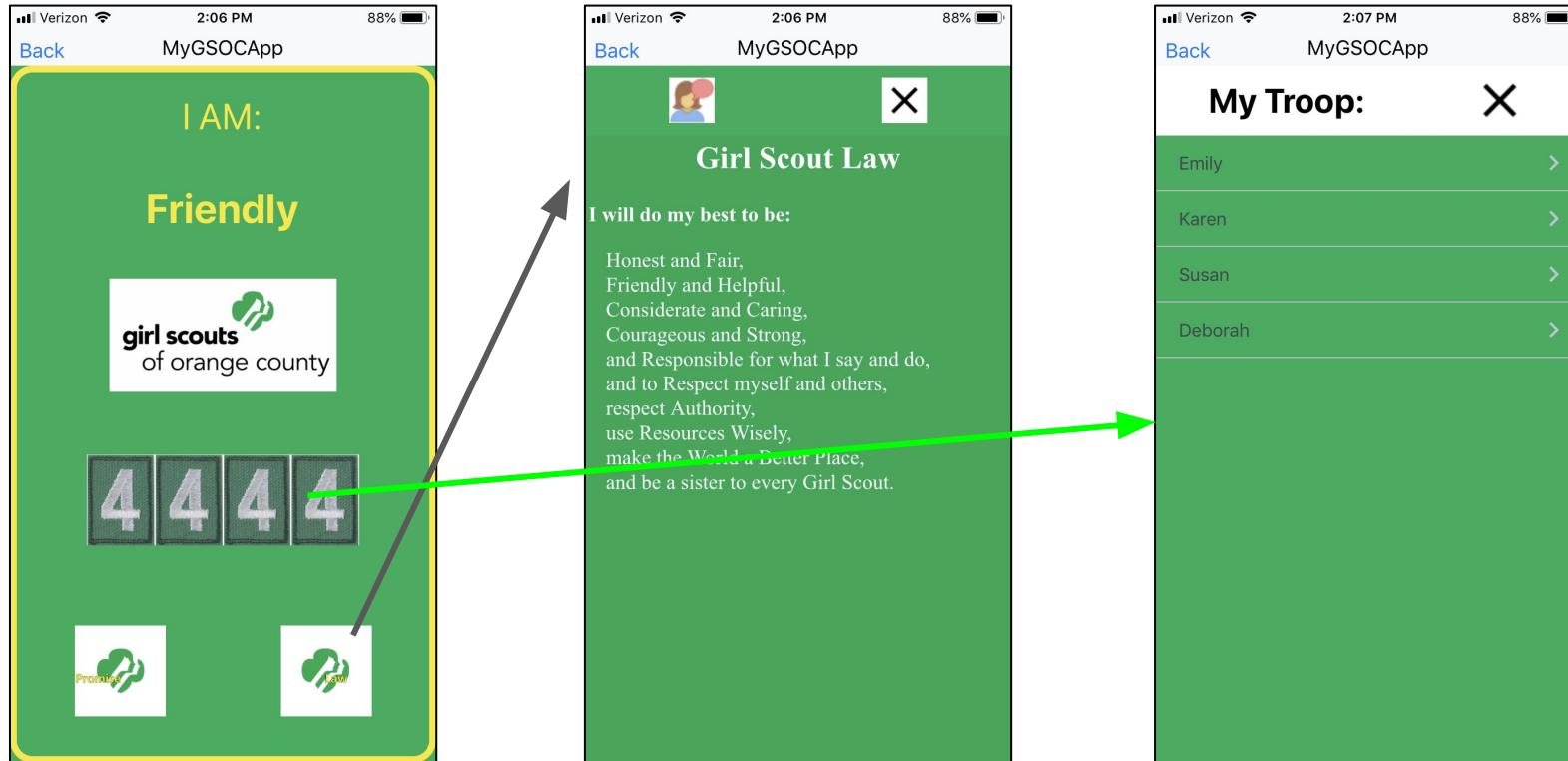
Select Blocks for scnTroop, then add a Click event block for our close button (btnTroopClose), so we can navigate back to scnHome when our scnTroop's close button is clicked.



The screenshot shows the thunkable app interface. The top navigation bar includes 'Live Test', 'Share', 'Make Copy', 'Download', 'Publish', 'Help', 'Community', and 'Account'. Below the navigation is a toolbar with icons for 'Public', 'Blocks', and 'Design'. The main workspace is titled 'scnTroop'. On the left, a sidebar lists categories: Control, Logic, Math, Text, Lists, Color, Device, Objects, Variables, Functions, sndLaw, sndPromise, lblTroopMyTroop, btnTroopClose, rowTopTroop, lstTroop, scnTroop, and Any Component. A yellow arrow points from the 'Control' category to a selected block: 'when btnTroopClose Click do [navigate to scnHome]'. The right side of the screen features a vertical toolbar with icons for zoom, plus, minus, and trash.

# Guided Design - viewing on our phone

Open the Thunkable Live app on your phone, then click the Live Test button at the top of the Thunkable X development environment. You should be able to navigate through your screens now!



# Challenges for extending your app ...

---

Add a note taking page to capture ideas for your Bronze, Silver or Gold Award projects.

Add a timer component and a label to display the timer countdown. Help your troop leader keep your meetings running on time!

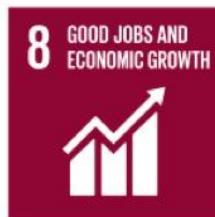
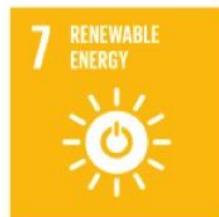
Create a “photos” page and add a camera and photo library component to capture photos taken at Girl Scout Events.

Create a “compass” page and use the magnetometer component and a graphical asset from [google.com/images](http://google.com/images) to create a compass that you can use for hiking at camp.

Find more assets! [google.com/images](http://google.com/images) and [material.io](http://material.io) websites are good sources of pictures you could use for button backgrounds.

# Brainstorming App ideas

Let's discuss app ideas that could address a problem in society, in our neighborhood, or in our school. Here are some "global goals" to consider to help you get started!



# Technovation Girls and the Congressional App Challenges

---

## ***Did you enjoy today's event?***

The Girl Scouts would like to support teams to enter the Technovation Girls (Feb/Mar 2020) and the very similar Congressional App (Oct/Nov 2020) app development challenges!

Here are the event links:

<https://technovationchallenge.org>

<https://www.congressionalappchallenge.us>

You must be in junior or senior high school to enter these two competitions. Since the event requirements are very similar, you can enter the identical project in both. There isn't an entry fee, and you'll have a chance to travel and win prizes for your team if your app is accepted!

**If you're not yet in junior high school, please enter the "Coolest Projects" event in March 2020:**

<https://coolestprojects.org/usa/>

**We can help you connect with mentors that can assist your team!**