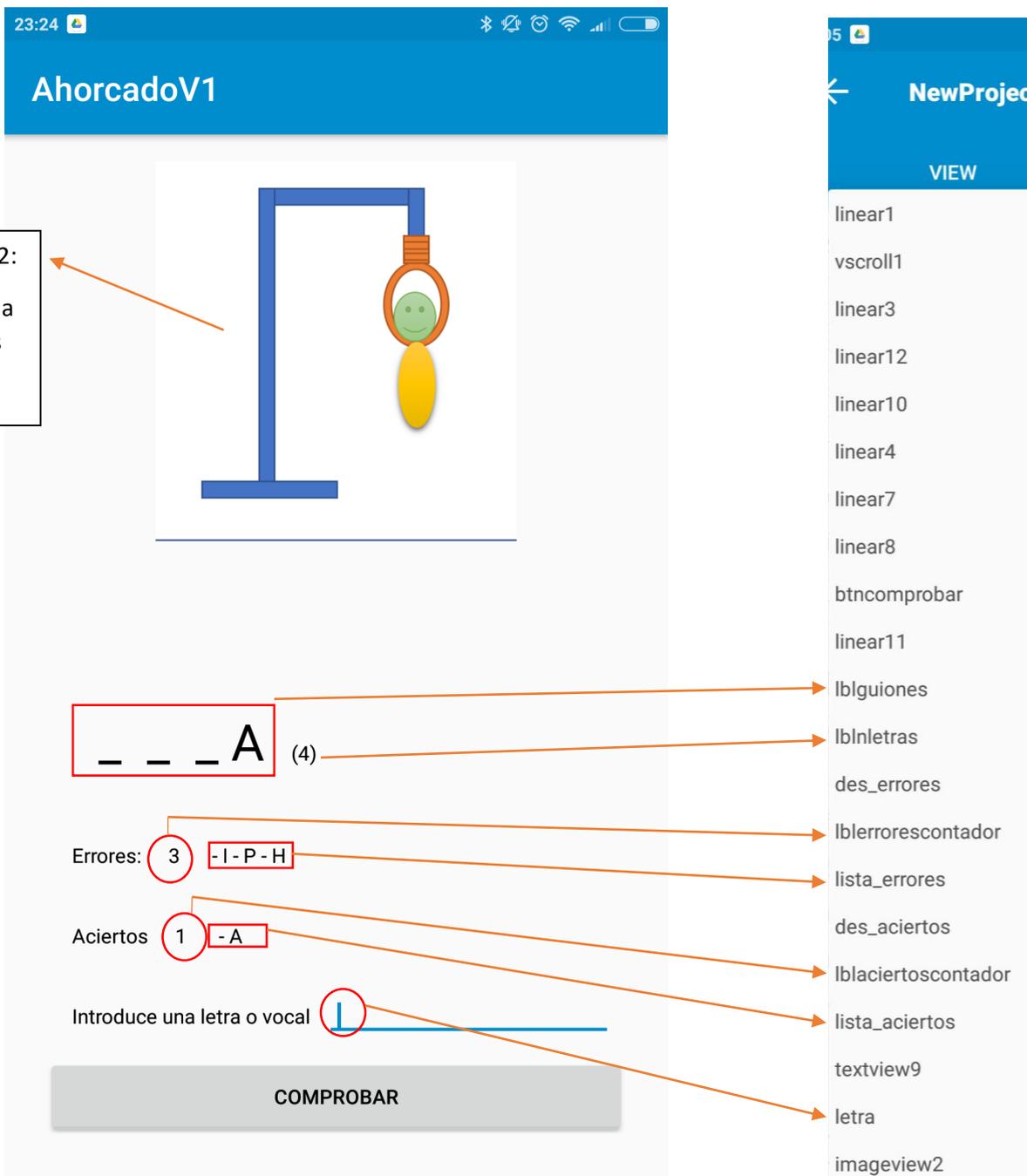


En esta actividad se pide realizar el juego del “ahorcado” para Android, usando nuestro teléfono móvil con Sketchware.

La idea principal de esta actividad es que te acostumbres a este entorno y pruebes los diferentes elementos que nos proporcionan: lista, variables, vistas, bloques,etc.

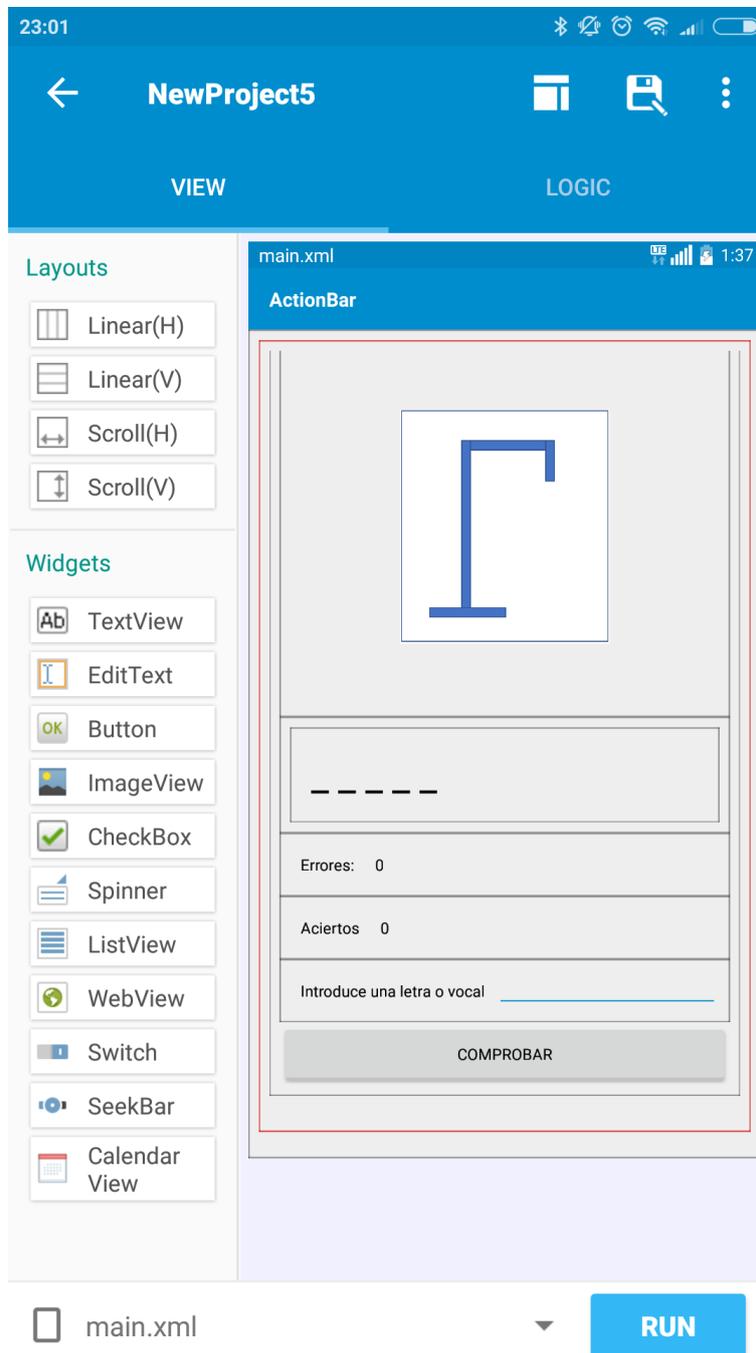
Siempre se ha de empezar por el diseño o la vista; se podría tener una apariencia distinta y se puede organizar de otra forma este juego, esta guía es solo una recomendación.

Vista el proyecto, ¿qué es lo que va a ver el usuario?, mira un ejemplo de la pantalla que queremos (recuerda que las imágenes las tienes en la web para descargar):



Verás que están puestos los elementos más importantes de la aplicación, el resto son disposiciones o texto para describir lo que va a poner, el cual no cambia en el programa (por ejemplo la palabra errores, aciertos, el texto del botón comprobar...etc)

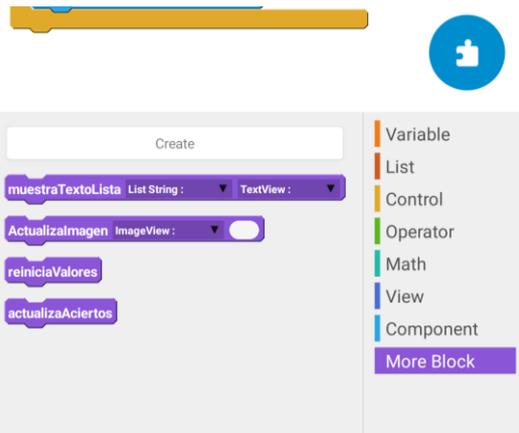
La pantalla que tendríamos que tener en el view y la disposición de los elementos podría ser esta:



Ahora, voy a ir a la lógica, donde debemos de “darle vida” al diseño, es decir, sino agregamos lo bloques, no tendremos nada. Para ello, nos vamos a ayudar de varios elementos y variables que podremos crear antes de empezar:

Estos bloques añadidos los vamos a usar para facilitar algunas funciones o trabajos en la aplicación. También se pueden usar para simplificar las pantallas.

Aceptan variables, y debemos de usar el que corresponda, ¡Cuidado! Es fácil equivocarse de tipo y no nos valdría de nada. En concreto aquí usamos en la primera: list String, TextView, e la segunda un ImageView, un contador (para relacionar los errores con la imagen a cargar). El resto no llevan parámetros.



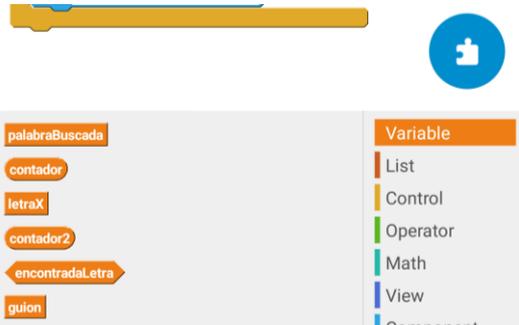
palabraBuscada: Va a almacenar la palabra que tiene que adivinar el usuario.

contador y contador2: lo usaremos varias veces para hacer cuentas en la aplicación para saber los índices y demás.

letraX: Es la letra que inserta el usuario para ver si está o no dentro de la palabraBuscada

guion: es un carácter que simula los guiones en concreto sería un espacio, guión bajo y otro espacio

“ _ “



listaPalabras: Va a ser un listado de palabras para adivinar, se han puesto algunas de prueba pero pueden añadirse muchas más.

erroresUsuario: Va a ser el listado de letras erróneas que ha metido el usuario.

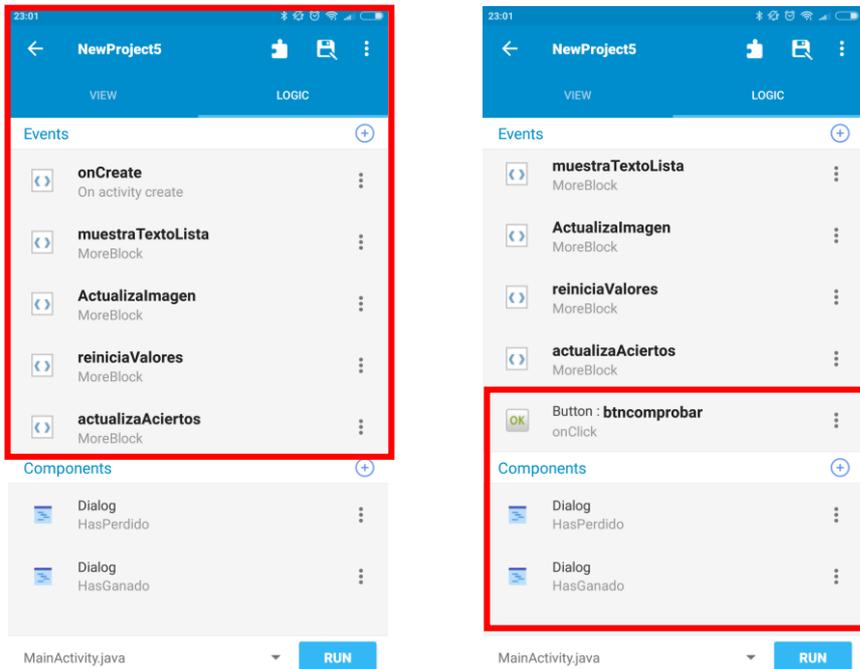
aciertosUsuario: Va a ser el listado de letras erróneas que ha metido el usuario.

palabraBusList: La palabra buscada la separamos en una lista para luego facilitar las búsquedas. Pepe, quedaría como [P],[e],[p],[e]

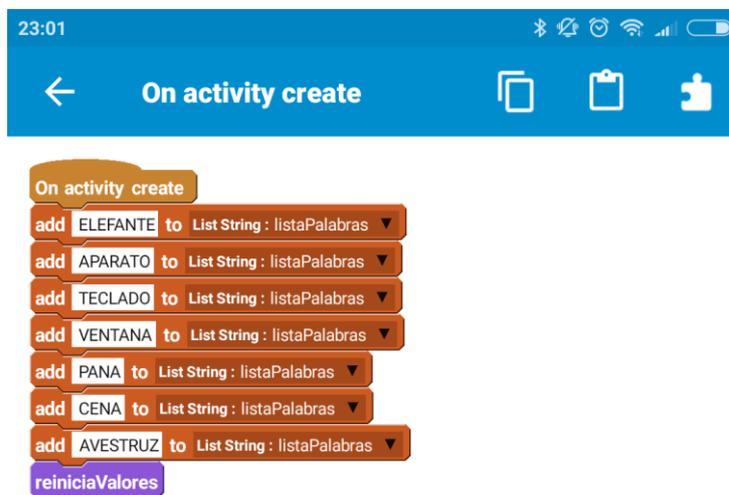
lbltextoguiones: es la relación de las palabras que ha acertado para tener el control de las que nuestro o no.



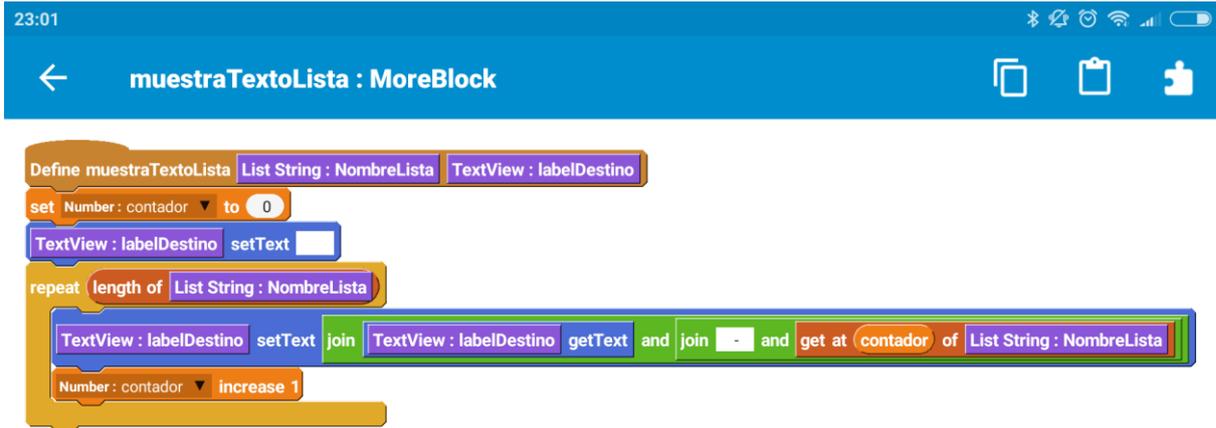
Ahora, tendríamos que tener los siguientes elementos en la pantalla inicial de bloques. Como no cogía en la misma captura he realizado dos, para que no te confundas y entiendas que puede haber varios eventos que no se vean en la pantalla inicial:



Lo primero, en el onCreate vamos hacer las cosas básicas, iniciar parámetros, crear listas...etc.:

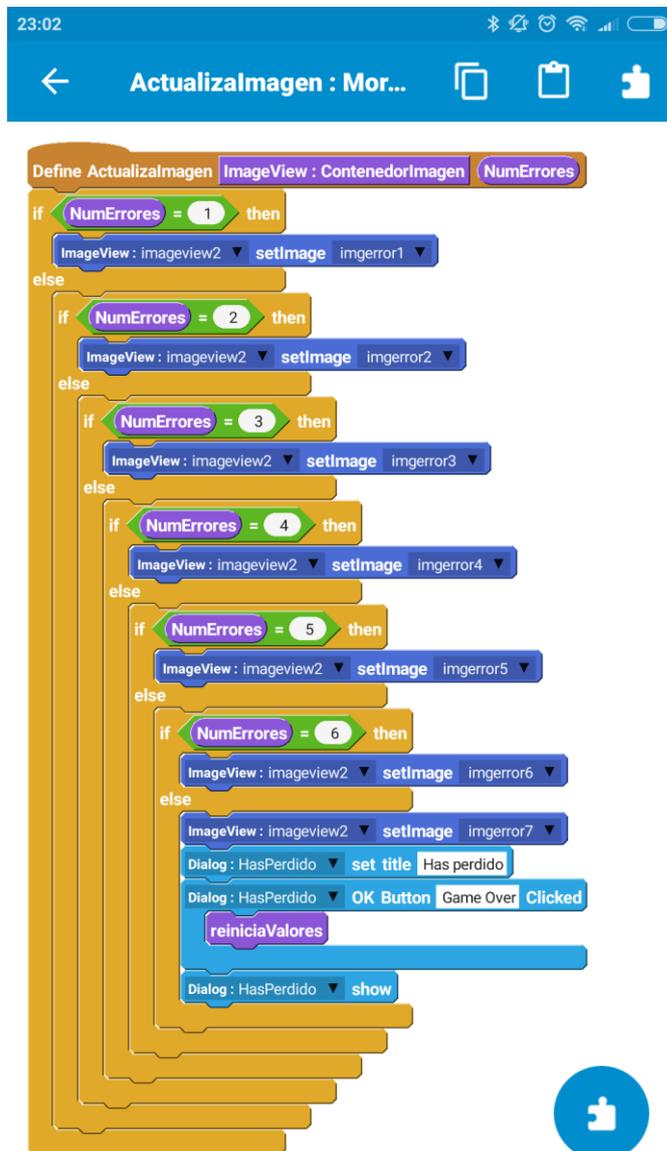


El bloque que nos hemos creado en “more Blocks” tenemos que decirle ¿qué vamos hacer en el, ya que este bloque es nuestro nos toca definir las acciones que nos interese.



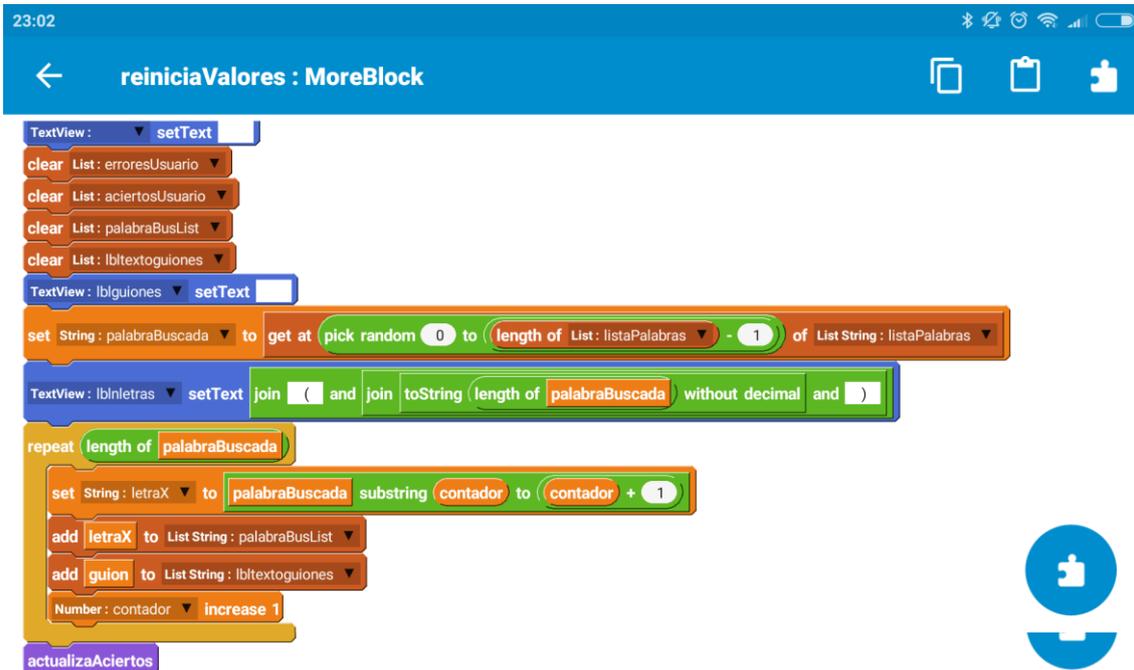
```
Define muestraTextoLista List String : NombreLista TextView : labelDestino
set Number : contador to 0
TextView : labelDestino setText 
repeat length of List String : NombreLista
  TextView : labelDestino setText join TextView : labelDestino getText and join - and get at contador of List String : NombreLista
  Number : contador increase 1
```

El bloque nuestro “Actualizalmagen” lo que va a hacer es, ¿cuántos errores lleva el usuario? Pues dependiendo del número de errores que lleve, le pongo una imagen u otra. Recuerda que las imágenes están en la web para descargarlas. Además si hemos llegado a última imagen, es que el usuario ha perdido, por lo que aprovechamos para informar y reiniciar todo.



```
Define Actualizalmagen ImageView : ContenedorImagen NumErrores
if NumErrores = 1 then
  ImageView : imageview2 setImage imgerror1
else
  if NumErrores = 2 then
    ImageView : imageview2 setImage imgerror2
  else
    if NumErrores = 3 then
      ImageView : imageview2 setImage imgerror3
    else
      if NumErrores = 4 then
        ImageView : imageview2 setImage imgerror4
      else
        if NumErrores = 5 then
          ImageView : imageview2 setImage imgerror5
        else
          if NumErrores = 6 then
            ImageView : imageview2 setImage imgerror6
          else
            Dialog : HasPerdido set title Has perdido
            Dialog : HasPerdido OK Button Game Over Clicked
            reiniciaValores
            Dialog : HasPerdido show
```

Hemos hablado varias veces de nuestro bloque “reiniciaValores” ¿qué hacemos en él? Piensa que tenemos que elegir otra palabra de lista, ponemos los contadores, listas y label como al inicio..etc.

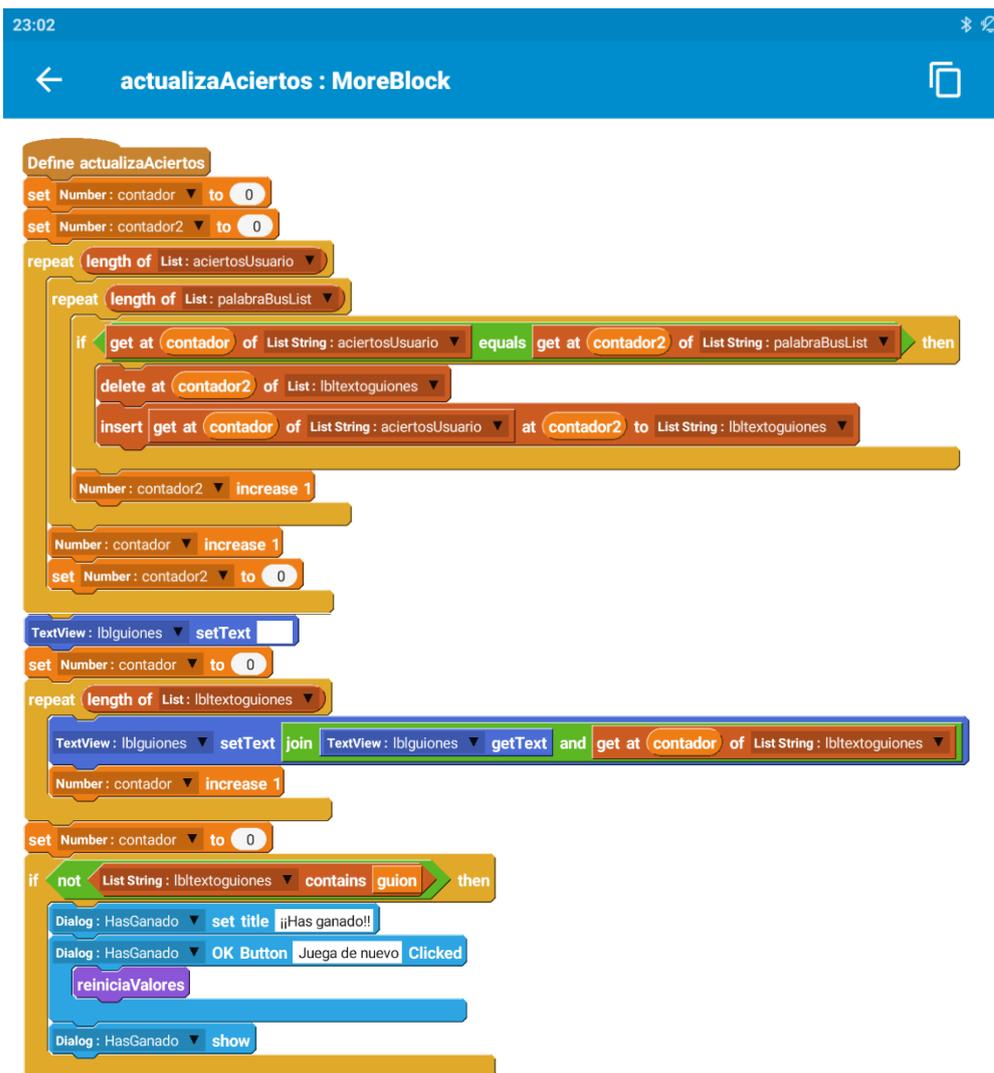


Luego tendríamos otro bloque nuestro que es **actualizaAciertos**. Esto se ejecuta para varias cosas, poner los guiones y letras en caso que el usuario haya acertado. Se pone al inicio, porque para optimizar, usamos el mismo código que genera los “_” de las letras y agrega los aciertos. Es decir, aquí no habría ningún acierto, por lo que pondría el largo del texto con guiones.

Además, en este bloque hacemos una cosa, si estamos comprobando la letra introducida por el usuario y ha finalizado todas (es decir no hay guiones en la palabra) es que ha terminado, así que mostramos mensajes y volvemos a reiniciar valores.

Otro detalle que hay que fijarse aquí es que vamos actualizando las listas y poniendo en aquellas letras que haya acertado, ponemos el valor correspondiente.

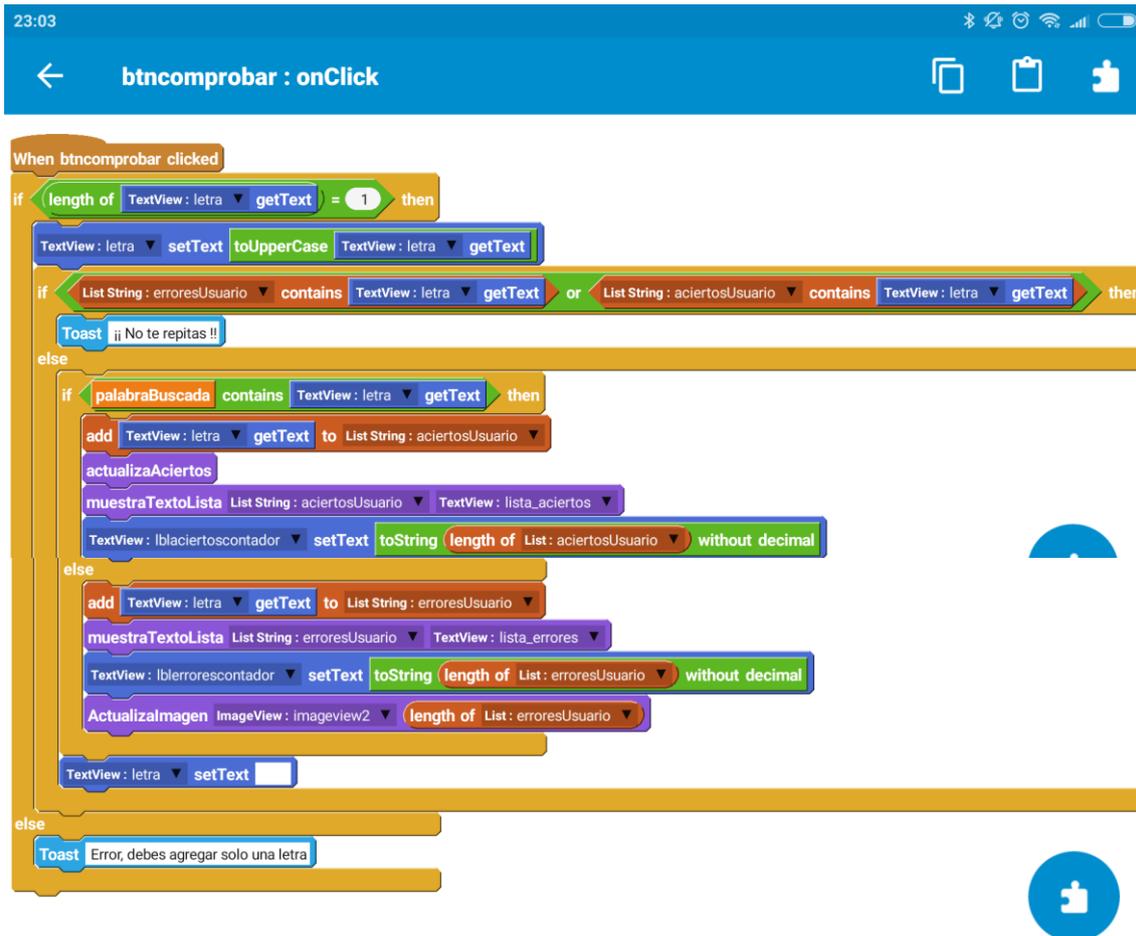
Para que se vea más fácil, y de un solo bloque, podrás ver en la siguiente imagen el código correspondiente:



Debes tener paciencia para hacer bloques grandes, y bueno, si dispones de una Tablet será más sencillo, también puedes ir por partes. Es decir, ve montando los bloques poco a poco, la idea es que cojas soltura de esto, así que no te desanimes, que luego el resultado merece la pena.

Por último, y no menos importante vamos a ver ¿qué hacemos cuando el usuario ha pulsado sobre el botón COMPROBAR?. Aquí tendremos que ir creando las listas, comprobar si la letra introducida está, o no está, y dependiendo de eso, hacer una cosa u otra.

Vuelvo a poner la imagen en la siguiente página para que se vea mejor y uniré las imágenes para que te sea más sencillo reproducirlas.



Bien, con todo esto tienes una gran ayuda, solo te falta reproducirlo por tu parte.

Tendrás que hacer la parte visual, añade las disposiciones o layout que creas convenientes (horizontales o verticales)

Agrega los componentes (dialog) uno para decir que ha ganado y otro para decir que ha perdido. (En verdad se puede usar el mismo, pero bueno, si te sobra tiempo lo pruebas)

Para agregar los bloques, debes estar dentro de la lógica, dentro de algún evento, la primera vez tendrás que estar dentro del onCreate o del onClick del botón. Y luego buscar "More block"

Ve poco a poco y ya verás como todo sale bien, al principio irás más despacio porque no estamos acostumbrados a mover los objetos, pero luego irá todo mejor, así que ánimo y cualquier cosa ponte en contacto con tu tutor.