

MEMOR Y FI



Realizado por:

→ Ariadna Amate

→ Naiara Cano

INDEX

Nombre del proyecto.....	2
Número de pantallas.....	2
¿Por qué hemos decidido este juego?.....	2
Página principal.....	2
Designer (Diseño).....	2
Blocks (Bloques).....	3
Página de “Inicio”	3
Designer (Diseño).....	3
Blocks (Bloques).....	5
Página de “Instrucciones”	5
Designer (Diseño).....	5
Blocks (Bloques).....	7
Página de “Registro” y “Login”	7
Designer (Diseño).....	7
Blocks (Bloques).....	9
Página de “Niveles”	10
Designer (Diseño).....	10
Blocks (Bloques).....	11
Niveles (fácil, normal, difícil, extremo).....	12
Designer (Diseño).....	12
Blocks (Bloques).....	14
Página de “Sonidos”	17
Designer (Diseño).....	17
Blocks (Bloques).....	19

(La documentación está en Castellano, ya le comentamos esto al Toni)

Nombre del proyecto

Memory F1

Número de pantallas

11 pantallas

¿Por qué hemos decidido este juego?

Principalmente hemos elegido el juego de "Memory", ya que es un juego bastante divertido y entretenido para todas las edades. La combinación de la Fórmula 1, que es algo que a ambas nos encanta y apasiona, con el desafío de recordar las ubicaciones de las cartas, puede hacer que la experiencia de juego sea muy atractiva, además nos ha ayudado a crear el juego con muchas más ganas. Al adaptar el juego de "Memory" a esta temática, se puede crear un juego que sea atractivo para los fanáticos de la F1 y que les permita disfrutar de su interés mientras juegan.

Además hemos añadido una opción de sonidos, que al dar clic en la imagen de alguno de los pilotos, te dice una palabra, que al decirla con la ayuda del "Text to speech", te responde con unos 5-6 segundos de canción o sonido relacionado con esa persona.

Página principal

Designer (Diseño)

Componentes utilizados:

- 1 Imagen
- 1 Reloj

Antes de empezar y añadir todos los componentes necesarios, editaremos la pantalla a nuestro gusto. Nosotras hemos cambiado el fondo a color negro, y desactivamos tanto el título y la barra de estado para que no sea visible.

☐ Show Options Menu

☐ Show Status Bar

Background Color
● #000000FF

Seguimos con los componentes y empezaremos con la imagen, en este caso únicamente pondremos el logo de F1, desactivaremos la opción de "Clickeable" y tanto de ancho como de alto lo pondremos en "Fill Parent" y así se ajustara en toda la pantalla.

Logo Properties

Common properties

☐ Clickable

Height
Fill parent

Width
Fill parent

Picture
F1logo.png

Rotation Angle
0.0

☐ Scale Picture To Fit

☒ Visible

Reloj Properties

Common properties

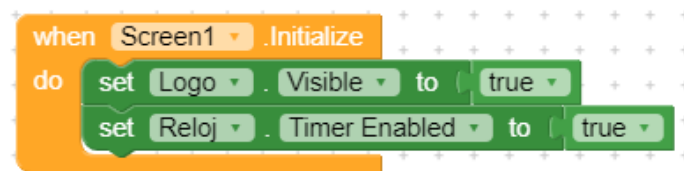
- ☒ Timer Always Fires
- ☒ Timer Enabled

Timer Interval
3000

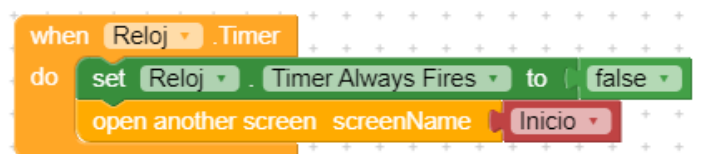
El reloj lo utilizaremos para que la pantalla dure “X” segundos activa, activaremos las opciones “*Temporizador siempre dispara*” y “*Temporizador habilitado*”, para que no dure mucho le colocaremos 3000 ms de intervalo (en los bloques lo configuraremos).

Blocks (Bloques)

Cuando se inicializa la pantalla “*Principal*”, nuestro código establece la visibilidad del logotipo a “*true*” y activa el temporizador del reloj. Esto significa que el logotipo será visible en la pantalla y que el temporizador del reloj comenzará a funcionar tan pronto como la pantalla se cargue. Al hacer esto, aseguramos que nuestra aplicación tenga una apariencia visual atractiva.



Cuando se activa el temporizador del reloj, nuestro código establece “*Timer Always Fires*” en “*false*” y abre la pantalla llamada “*Inicio*”. Al hacer esto, el temporizador del reloj solo se activará una vez y no se repetirá constantemente.



Página de “Inicio”

Designer (Diseño)

Componentes utilizados:

- 2 Label
- 2 Espacios
- 2 Botones

Antes de empezar y añadir todos los componentes necesarios, editaremos la pantalla a nuestro gusto. Nosotras hemos cambiado el fondo y le hemos colocado una imagen a juego con el tema escogido para el memory, y desactivamos tanto el título y la barra de estado para que no sea visible.



Comenzaremos colocando los espacios, el primero en la parte alta de la pantalla para dejar algo de espacio con el título y que así no quede tan pegado, a este le pondremos una altura de 60 px y la anchura la dejaremos en automática. El segundo espacio estará justo por debajo del título y a este le colocaremos una altura de 450 px y también la anchura se quedará en automática.

Espacio Properties	Espacio2 Properties
Common properties	
Height: 60px	Height: 450px
Width: Automatic	Width: Automatic

Memory Properties	Label Properties
Font Size: 35	Text: MEMORY
Font Typeface: Default	Text Alignment: Left : 0
Height: Automatic	Text Color: #EF1D16FE
Width: Automatic	Advanced properties
	Font Typeface Import: Formula1-Wide.ttf

Ahora colocamos dos etiquetas "label" justo debajo del primer espacio, en el primero pondremos "Memory" y el segundo justo debajo "F1". Tanto la altura como la anchura de ambos se quedará en automática, la medida será de 35 y le pondremos un tipo de letra que hemos subido anteriormente, típico de la Fórmula 1, por último el color será rojo, a juego con el logo y la imagen puesta de fondo.

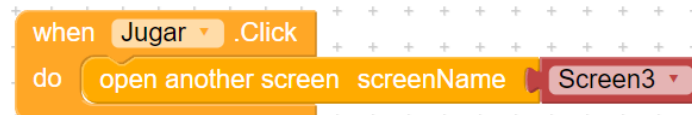
Para acabar con esta pantalla colocaremos los dos botones, uno que tendrá el nombre de "Jugar" y nos llevará a la pantalla de login, le cambiaremos el color a rojo, la letra será la que subimos previamente, su medida de 25 y el color blanco, su altura será de 50 px y la anchura 250 px, para acabar con este le pondremos forma redondeada. El segundo botón tendrá el nombre de "Como jugar" y nos llevará a la pantalla de instrucciones, este boton estara justo debajo del anterior y tendra las mismas propiedades, lo unico que lo distinguira sera el color, este sera negro y el color de la letra blanco.

Jugar Properties
Common properties
Background Color: #EF1D16FE
<input checked="" type="checkbox"/> Enabled
<input type="checkbox"/> Font Bold
<input type="checkbox"/> Font Italic
Font Size: 25.0

Jugar Properties	Como_Jugar Properties
Height: 50px	Height: 50px
Width: 250px	Width: 250px
Shape: Rounded	Shape: Rounded
Text: Jugar	Text: Como jugar
Text Alignment: Center : 1	Text Alignment: Center : 1
Text Color: #FFFFFFF	Text Color: #000000F
Advanced properties	Advanced properties
<input checked="" type="checkbox"/> Border Shadow	<input type="checkbox"/> Border Shadow
Font Typeface Import: Formula1-Wide.ttf	Font Typeface Import: Formula1-Wide.ttf

Blocks (Bloques)

Cuando hacemos clic en el botón “Jugar”, nuestro código abre otra pantalla llamada “Register”. Esto nos permite organizar nuestra aplicación en varias pantallas y guiar al usuario a través de diferentes secciones de forma intuitiva y sencilla.



Cuando hacemos clic en el botón “Como Jugar”, nuestro código abre la pantalla llamada “Instrucciones”. Esta pantalla nos ofrece la oportunidad de proporcionar a los usuarios información detallada sobre cómo jugar a Memory en nuestra app. Al colocar las instrucciones en una pantalla separada, evitamos que la pantalla principal se vea abarrotada.



Página de “Instrucciones”

Designer (Diseño)

Componentes utilizados:

- 7 Botones
- 4 Espacios
- 1 Tabla
- 1 Imagen

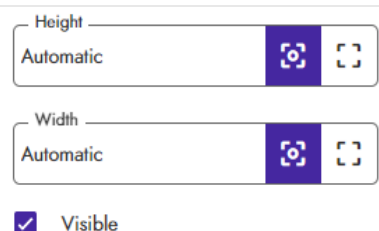
Antes de empezar y añadir todos los componentes necesarios, editaremos la pantalla a nuestro gusto. Nosotras le hemos puesto la misma imagen de fondo que la anterior pantalla, desactivamos tanto el título y la barra de estado para que no sea visible.



Comenzaremos colocando los espacios, el primero en la parte alta de la pantalla para dejar algo de espacio con el título y que así no quede tan pegado, a este le dejaremos tanto la altura como la anchura en automático. El segundo espacio estará justo por debajo del título y a este también le dejaremos la altura y anchura en automático. El tercer espacio lo pondremos en una tabla que luego crearemos para poner una foto, a este solo le pondremos una anchura de 60 px. Y el último nos servirá para separar la tabla y el último

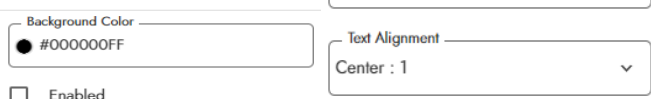
Espacio Properties

Common properties



Como_Jugar Properties

Common properties



botón, le pondremos una altura de 50 px.

A continuación pondremos los botones, en este caso solo los usaremos para colocar texto, también podríamos utilizar el componente de etiqueta “*label*”, pero a nosotras se nos hace más sencillo con botones, así que los usaremos. El primer boton sera para el titulo, le cambiaremos el color a negro, la letra será la que subimos previamente, su medida de 20 y el color blanco, su altura será automatica y la anchura la pondremos en “*Fill Parent*”, para acabar desactivaremos la opcion de “*Enabled*”.

Norma1 Properties

Common properties

Background Color #F911038A

☐ Enabled

☐ Font Bold

☐ Font Italic

Font Size 23

Font Typeface Font Awesome

Text Color #FFFFFF

Los otros 5 botones los utilizaremos para poner las normas o instrucciones, el color será rojo, la letra será la “*Serif*”, su medida de 23 y el color blanco, su altura varia depende la cantidad de letra que tenga, entre 60 px y 100 px, la anchura la colocaremos en “*Fill Parent*”. El último botón nos servirá para volver a la pantalla de inicio y así hacer click en “*Jugar*”, tendra las mismas propiedades que el boton de titulo, solo que este sera

Width Fill parent

Height 80px

de color

blanco, con la letra de color negro.

La tabla la colocaremos justo debajo de todos los botones de las normas, esta tendrá 3 columnas y 2 filas, con una altura de 100 px, aunque eso será irrelevante, ya que solo la utilizaremos para colocar la imagen en el centro.

Tabla Properties

Common properties

Columns 3

Height 100px

Width Fill parent

Rows 2

Height 100px

Width 130px

Picture FIA.png

Seguimos con la imagen que colocaremos en la tabla como bien he dicho, esta será una foto del logo de la “*FIA*” (Federación Internacional del Automóvil, es mundialmente conocida por regular las competiciones de automovilismo más importantes del mundo, pero su ámbito de aplicación incluye todos los aspectos del automóvil, las carreteras, la movilidad, el medio

ambiente o seguridad vial), tendrá una altura de 100 px y una anchura de 130 px.

Blocks (Bloques)

Cuando clickeamos el botón de “Inicio”, nos llevará a la pantalla de inicio de vuelta.



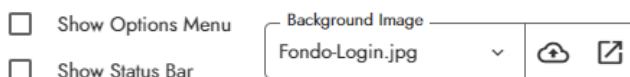
Página de “Registro” y “Login”

Designer (Diseño)

Componentes utilizados:

- 5 Espacios
- 4 Botones
- 2 Recuadros Blancos
- 2 Cajas de Texto
- Base de Datos
- Notificación

Antes de empezar y añadir todos los componentes necesarios, editaremos la pantalla a nuestro gusto. Nosotras le hemos puesto una imagen de fondo, desactivamos tanto el título y la barra de estado para que no sea visible.



Comenzaremos colocando los espacios, el primero en la parte alta de la pantalla para dejar algo de espacio con el título de usuario y que así no quede tan pegado, a este le dejaremos tanto la altura como la anchura en automático. El segundo espacio estará justo por debajo del título y a este también le dejaremos la altura y anchura en automático. El tercer espacio lo pondremos debajo de la primera caja de texto.

Espacio Properties

Common properties



Espacio5 Properties

Common properties



El siguiente estará debajo del título de contraseña. Y el último nos servirá para separar el usuario y contraseña del botón para cambiar de pantalla y loguearse, le pondremos una altura de 340 px.

UsuarioTexto Properties

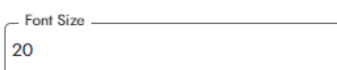
Common properties



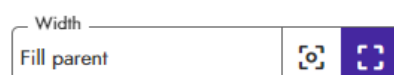
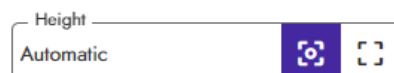
☐ Enabled

☐ Font Bold

☐ Font Italic



A continuación pondremos los botones, el primer botón será para el título de usuario, le cambiaremos el color a rojo, la letra será la que subimos previamente, su medida de 20 y el color blanco, su altura será automática y la anchura la pondremos en “Fill Parent”, para acabar desactivaremos la opción de “enabled”. Haremos lo mismo con el siguiente botón, solo que este será para el título de la contraseña. Seguimos



con el tercer botón, donde colocaremos un coche típico de la F1 y este será para registrar tanto el usuario como la contraseña, tendrá las mismas propiedades solo, que el tipo de letra será una diferente subida previamente y su medida sera de 33, ademas no desactivaremos la opción de “Enabled”. Para acabar con los botones, el último lo colocaremos en la parte baja de la pantalla y este será para cambiar a la pantalla de “Login”, por si ya estuviéramos registrados, sus propiedades seran como los botones de usuario y contraseña, solo que el boton sera blanco y la letra de color negro.

Font Size
18.0

Font Typeface
Default

Height
Fill parent

Width
Fill parent

Pondremos dos recuadros blancos junto a dos cajas de texto, debajo de los dos primeros botones, que usamos de título (usuario y contraseña), para poder escribir el usuario y la contraseña. La altura y anchura la pondremos en “Fill Parent” y la medida de la letra será de 18 px. En bloques configuraremos para que al darle al botón del coche si hay algún campo vacío no cambie de pantalla y notifique que falta un campo por rellenar.

En esta pantalla pondremos la base de datos la cual se encargará de guardar todos los usuarios y sus respectivas contraseñas. Lo configuraremos en el apartado de bloques.

Advanced properties

☒ Border Shadow

Font Typeface Import
Formula1-Wide.ttf

Base_de_datos Properties

Common properties

Namespace
TinyDB1

Por último colocaremos una notificación, que esta nos avisará, cuando el usuario ya esté registrado o cuando se queda algún espacio sin rellenar. También lo configuraremos en el apartado de bloques. Lo único que editaremos en las propiedades será el color de la letra a negro y la duracion de la notificacion que la cambiaremos a corta, para que no dure tanto.

Notifier Length
Short

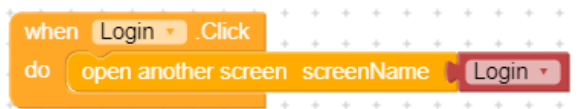
Text Color
● #000000FF

En la página de “Login” será exactamente igual que la de registro, lo único que cambiará será el botón de login, ya que este será para regresar a la página de registro. Además de que en bloques, al loguearte en esta página, inicias sesión y te llevará directamente a la página de niveles, para poder jugar.

Text
Registrar

Blocks (Bloques)

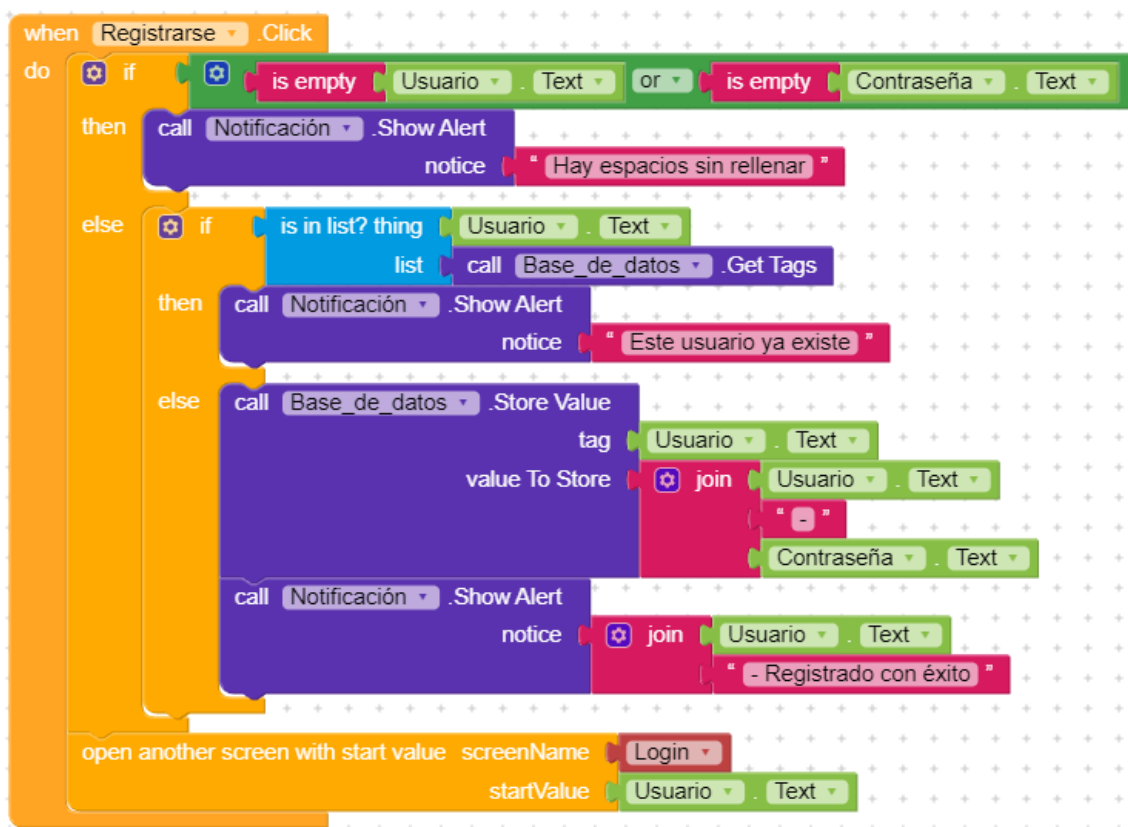
Para configurar el botón de login, para cuando el usuario hace clic en “Login”, nuestro código abre la pantalla llamada “Login”.



Si el usuario hace clic en “*Registrarse*”, se verifica si los campos de la caja de texto “*Usuario*” y “*Contraseña*” están vacíos. Si alguno de ellos está vacío, se muestra una alerta de notificación indicando que hay campos sin rellenar.

Si ambos campos están completos, se verifica si el nombre de usuario ya existe en la lista de usuarios almacenados en la base de datos. Si el nombre de usuario ya existe, se muestra una notificación indicando que el usuario ya existe.

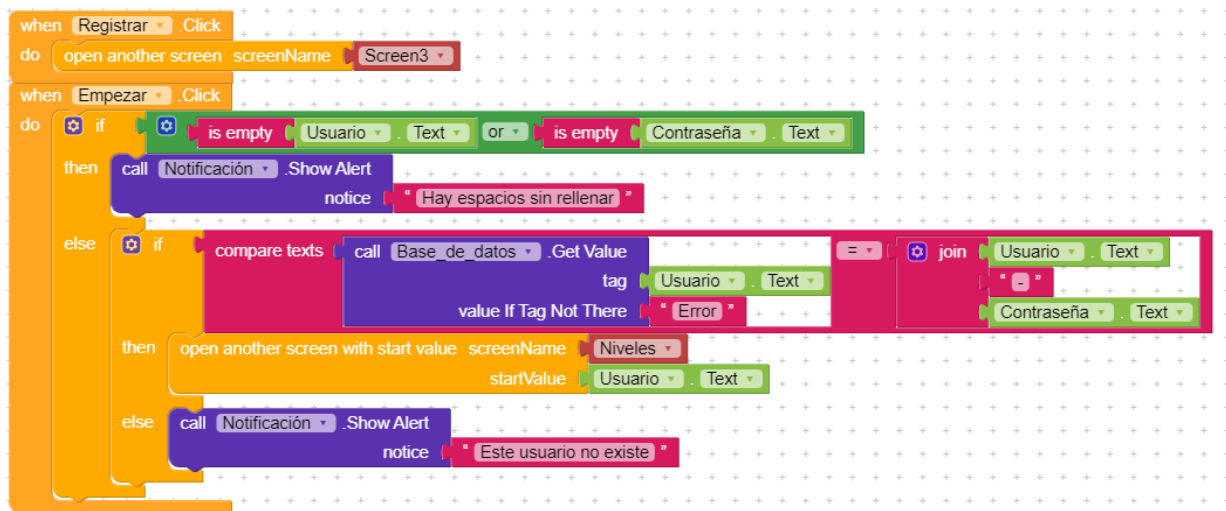
Si el nombre de usuario no existe, se almacena en la base de datos junto con la contraseña. Luego, se muestra una notificación indicando que el registro se ha realizado correctamente y se abre la pantalla “*Login*”, donde el usuario pondrá su nombre y contraseñas introducidos para así iniciar sesión.



Para configurar el botón de registrar, para cuando el usuario hace clic en “*Registrar*”, nuestro código abre la pantalla llamada “*Registrar*”.

Si el usuario hace clic en “*Loguearse*”, se verifica si los campos de la caja de texto “*Usuario*” y “*Contraseña*” están vacíos. Si alguno de ellos está vacío, se muestra una alerta de notificación indicando que hay campos sin rellenar.

Si ambos campos están completos, se abre la pantalla de Niveles para que el usuario pueda jugar libremente.



Página de “Niveles”

Designer (Diseño)

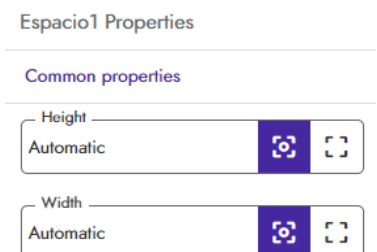
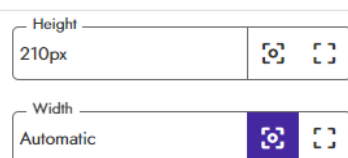
Componentes utilizados:

- 7 Espacios
- 1 Label
- 6 Botones

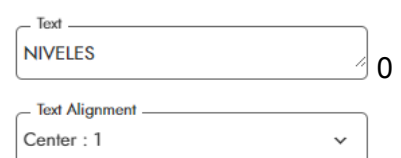
Antes de empezar y añadir todos los componentes necesarios, editaremos la pantalla a nuestro gusto. Nosotras le hemos puesto una imagen de fondo, desactivamos tanto el título y la barra de estado para que no sea visible.



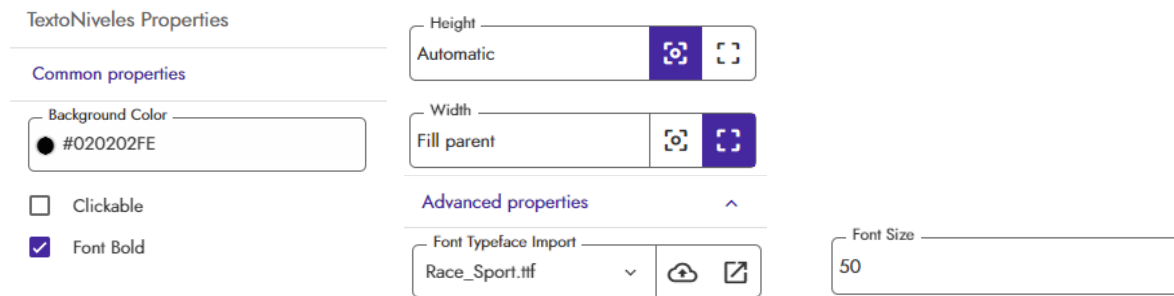
Comenzaremos colocando los espacios, todos estarán de altura y anchura en automático, ya que los utilizaremos para separar el label del título y los botones de los niveles. El último en cambio tendrá una altura de 210 px y se utilizara para separar todos los botones de niveles y sonido con el botón de “cerrar sesión”, que se quedara el parte baja de la pantalla.



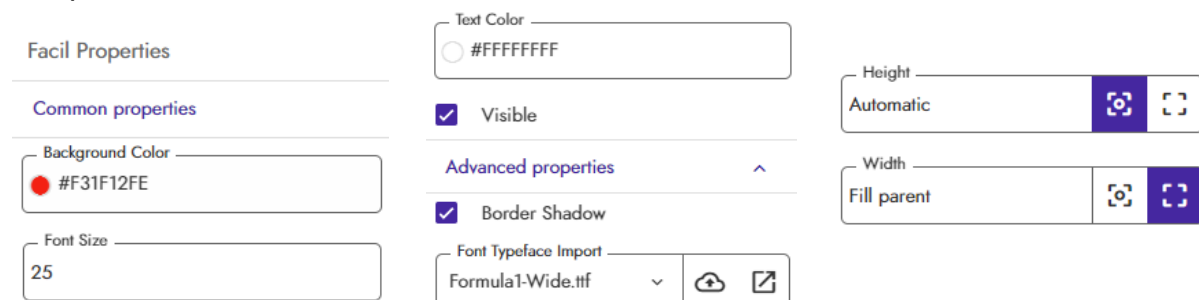
El único label se utiliza para el título, este tendrá el nombre de “Niveles”, su altura será automática y la anchura la pondremos en “Fill Parent”, el fondo sera color



negro, el tipo de letra subido anteriormente y la medida de 50, además activaremos la opción de “**Bold**” y su color será rojo.

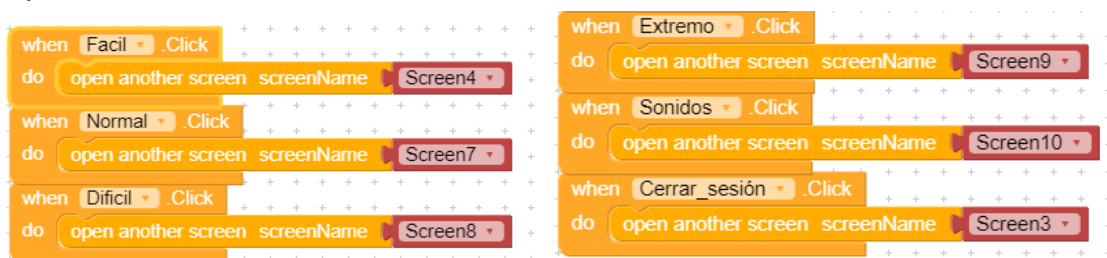


Para acabar colocaremos los botones de los diferentes niveles que habrá (fácil, normal, difícil, extremo), serán de color rojo, la fuente subida previamente, de color blanco y su medida será de 25, su altura se quedará en automático y la anchura la pondremos en “*Fill Parent*”. Como hemos dicho anteriormente todos estarán separados con un espacio. Los configuraremos en el apartado de bloques.



Blocks (Bloques)

Para configurar los botones de los distintos niveles, sonidos y cerrar sesión, para cuando el usuario hace clic en “*Login*”, nuestro código abre las pantallas específicas de cada uno.



Niveles (fácil, normal, difícil, extremo)

Designer (Diseño)

Componentes utilizados:

- 2 o 3 Espacios

- 1 Label
- 1 Tabla
- Entre 8 y 42 Botones (Depende del nivel)
- 1 Reloj
- 1 Notificación

Antes de empezar y añadir todos los componentes necesarios, editaremos la pantalla a nuestro gusto. Nosotras le hemos puesto una imagen de fondo, desactivamos tanto el título y la barra de estado para que no sea visible.

☐ Show Options Menu
 ☐ Show Status Bar

Background Image
 FondoNiveles.jpg

Comenzaremos colocando los espacios, ambos serán para separar el título tendrán una altura de 30 px y 70 px.

Espacio1 Properties

Common properties

Height
30px

Width
Automatic

Espacio2 Properties

Common properties

Height
70px

Width
Automatic

Agregamos una etiqueta “Label” para el título, configuramos el ancho del label como “Fill Parent”, sera de color negro, la letra subida anteriormente, la medida de 33 y el color blanco.

Texto Properties

Common properties

Background Color
#020202FE

Height
Automatic

Width
Fill parent

Text Color
#FFFFFF

☒ Visible

Advanced properties

Font Typeface Import
Race_Sport.ttf

Font Size
33

Para poder ordenar todos los botones, que serán cartas para nosotras, crearemos una tabla “table arrangement”, le pondremos 2 columnas y 3 filas al primer nivel, 3 columnas y 4 filas al segundo nivel, 4 columnas y 5 filas al tercer nivel, y por último 5 columnas y 8 filas al cuarto nivel. La altura y anchura de todas la pondremos en “Fill Parent”

Ahora añadimos los botones, que será lo más importante en estas pantallas, ya que simularán las cartas del juego de Memory. Dentro de la tabla colocaremos cada botón, y depende el nivel serán más grandes o más pequeños, ya que si hay más columnas y filas hay que dejar

Tabla Properties

Common properties

Columns
3

Height
Fill parent

Width
Fill parent

Rows
4

espacio para todos en la pantalla. Para que no todos sean del mismo color, hemos puesto los colores de las escuderías de F1, un color por botón en el caso del primer nivel, en los demás, al ser más botones lógicamente, hemos repetido los colores, también quitamos el texto de los botones para que estén vacíos y le ponemos la forma redondeada. Todo se configura en el apartado de bloques.

Carta5 Properties

Common properties

Background Color

☒ Enabled

☐ Font Bold

☐ Font Italic

Font Size

Text Color

Shape

Reloj Properties

Common properties

☒ Timer Always Fires

☒ Timer Enabled

Timer Interval

Necesitaremos un reloj, para controlar cuando los botones “*desaparecen*” (dejan de estar visibles) y esto aparenta que la carta se da la vuelta y aparece la imagen de alguno de los pilotos. Le ponemos un intervalo de 1000 ms, para que no sea ni muy largo ni muy corto. Además de todas las opciones activadas. Todo esto se configura en el apartado de bloques.

Para que todo sea más interactivo añadiremos una notificación que nos avisará cuando las dos cartas sean iguales, lo que significa que hemos encontrado la pareja, con la frase de “CHACHI!!!” y si en cambio no son iguales, que significa que no hemos encontrado la pareja, nos saldrá la frase de “NO CHACHI :C”. Todo esto se configura en el apartado de bloques.

Mensaje Properties

Common properties

Background Color

☐ Light Theme

☒ Linkify

Notifier Length

Blocks (Bloques)

Para empezar con este gran código, creamos unas variables de lista, que serán las siguientes:

- “*Cartas*”: Para almacenar los botones (cartas).
- “*BotonClick*”: Para almacenar el primer botón clickeado.
- “*BotonClick2*”: Para almacenar el segundo botón clickeado.

- "*Imágenes*": Para almacenar todas las imágenes disponibles.
- "*ImagenCartas*": Para almacenar las imágenes de las cartas que se mostrarán en el juego.

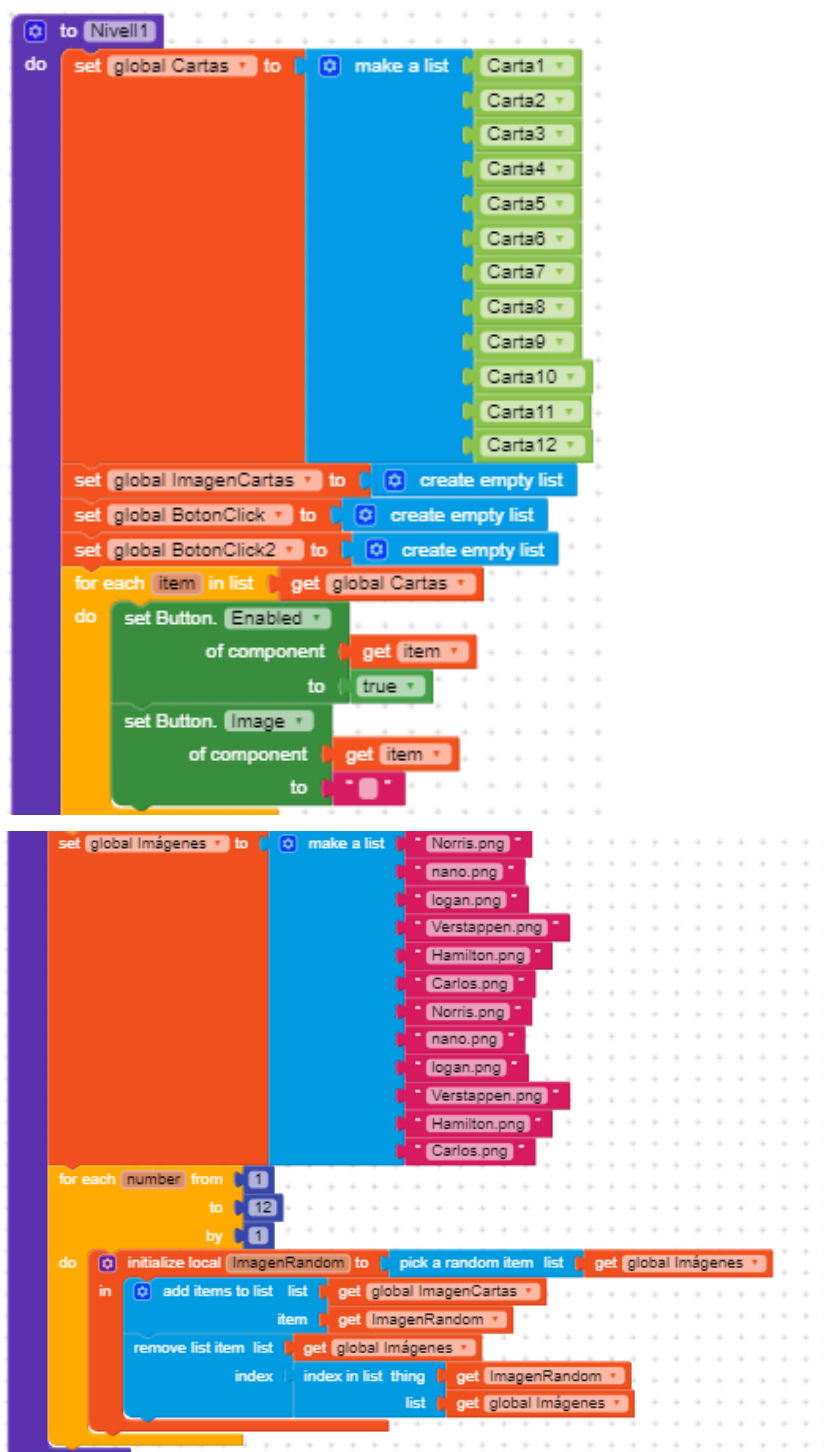


```

initialize global Cartas to create empty list
initialize global Imágenes to create empty list
initialize global ImagenCartas to create empty list
initialize global BotonClick to create empty list
initialize global BotonClick2 to create empty list
  
```

Creamos una Función con el nombre del nivel (nivel 1, 2, 3, 4), esta tendrá lo siguiente:

- Se añaden a la lista creada de "*Cartas*", todos los botones (cartas), uno a uno.
- Seguidamente se añaden las variables de lista vacía ImagenCartas, BotonClick, BotonClick2, creadas anteriormente.
- Llamamos a la global Cartas para dejar el botón en "*Enabled*" y la imagen con un texto vacío.
- Añadimos a la lista creada de "*Imágenes*", todas las imágenes, repetidas una vez.
- Y por último pedimos que se elija una imagen aleatoria de la lista para cada botón (carta), para que esto funcione se tendrá que hacer botón por botón.



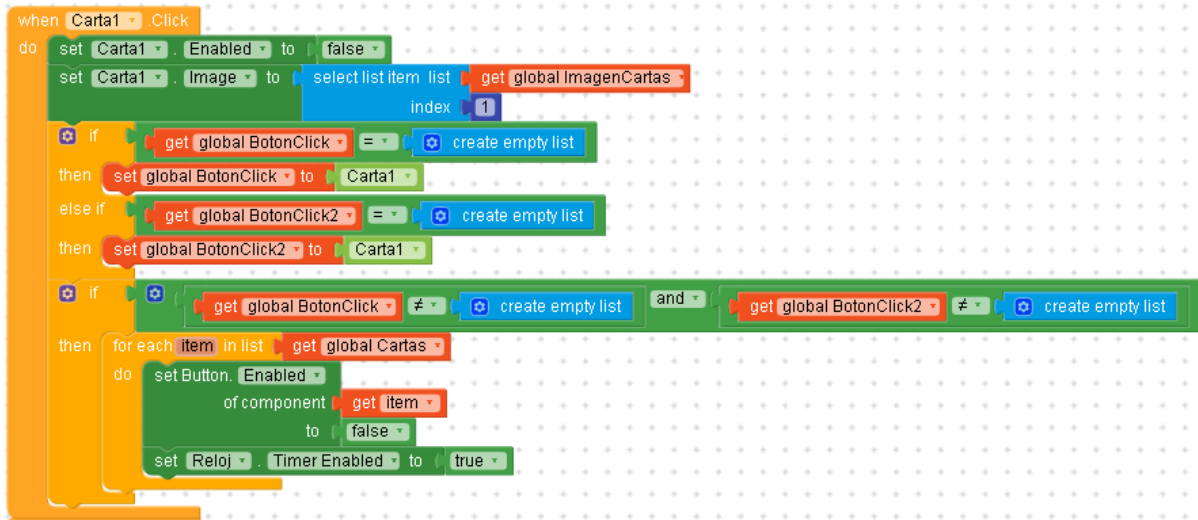
Al crear la función nos tocará llamarla así que cuando se inicialice la pantalla llamaremos a esta misma función.



Ahora haremos que funcione boton por boton, uno a uno, para eso hacemos lo siguiente:

- Cuando clickeamos cualquier botón este se desactiva (desaparecerá), se selecciona una imagen aleatoria y se pondrá como visible.

- Añadimos un if, si el botón es el primero en clickear, se pondrá una imagen, si es el segundo en clickear, se pondrá otra.
- Con otro if, activamos el reloj



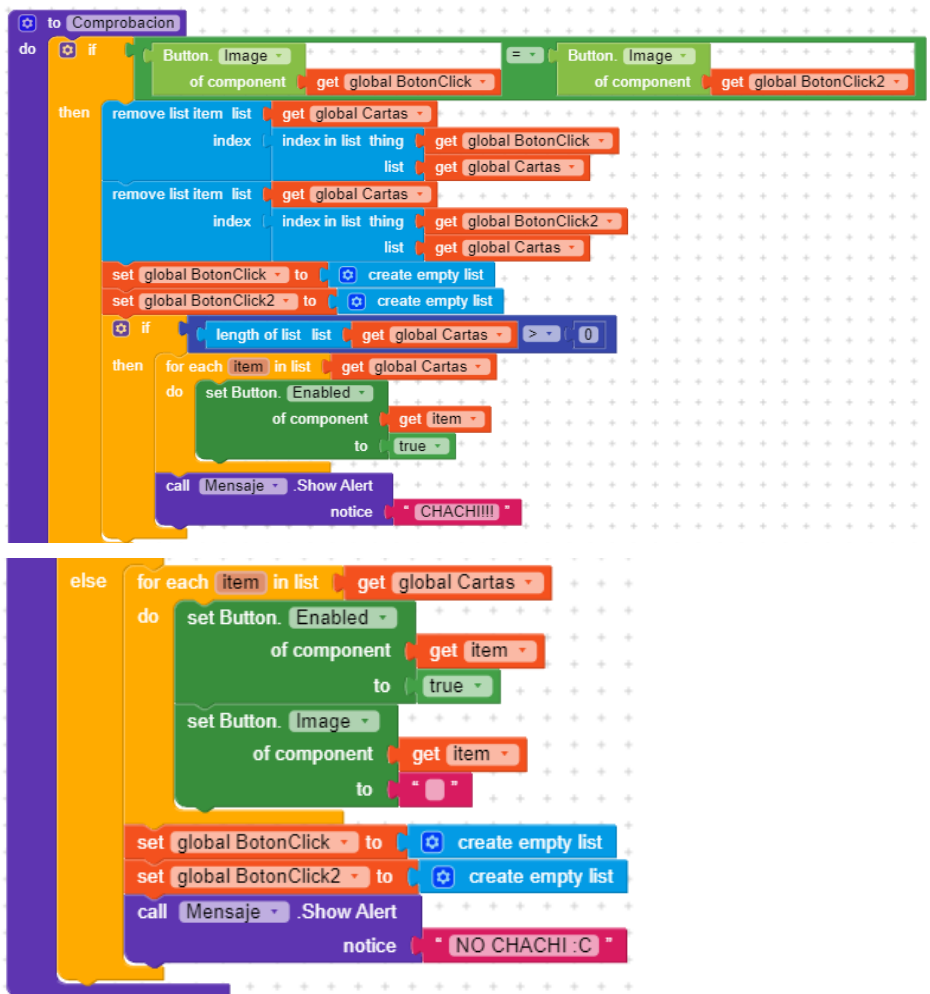
Por si acabamos la partida y nos apetece jugar de nuevo, sin la necesidad de tirar para atrás, creamos un botón de reinicio, para cuando clickeamos “Reseteear” llame a la función creada anteriormente “Nivel 1, 2, 3 o 4” y así reiniciar el juego



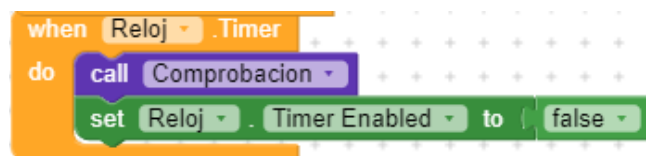
Para comprobar que las cartas son realmente iguales tenemos que crear otra función, a esta le ponemos el nombre de “Comprobación”. La cual comprueba con ayuda de un if si la imagen clickeada por primera vez es la misma a la de la segunda.

Si es así, elimina esta carta de la lista “Cartas”, para evitar que esta se repita, ya que la pareja está formada. Y además gracias a la notificación puesta anteriormente, nos envía un mensaje de “CHACHI!!!!”.

Si no es así, vuelve a aparecer el boton (se da la vuelta de nuevo la carta) y se sigue la partida, lo único que nos aparece la notificación de “NO CHACHI :C”



Y por último gracias al reloj llamamos a esta nueva función para que se deshabilite.



Página de “Sonidos”

Designer (Diseño)

Componentes utilizados:

- 3 Espacios
- 4 Botones
- 1 Tabla
- 9 Recuadros Blancos
- 9 Imagen
- 9 Sonidos
- 1 Reconocimiento de Voz
- 1 Texto a Voz

Antes de empezar y añadir todos los componentes necesarios, editaremos la pantalla a nuestro gusto. Nosotras le hemos puesto la misma imagen que en el fondo de “Niveles”, desactivamos tanto el título y la barra de estado para que no sea visible.

☐ Show Options Menu
 ☐ Show Status Bar

Background Image
 FondoNiveles.jpg

Comenzaremos colocando los espacios, los utilizaremos para separar el título y botones, todos estarán tanto de altura como de anchura en automático.

Espacio3 Properties

Common properties

Height
 Automatic

Width
 Automatic

Para los botones, utilizaremos el primero y tercero de información, en forma de label, el primero para el título, el tercero en cambio para informar a los usuarios que haciendo click en las imágenes se escuchara el nombre que tienen que decir para escuchar el sonido del piloto. En cambio el segundo botón que utilizaremos será justo para decir lo que escuchemos, al clicar en él activaremos el “text to speech”. Y el último botón será para poder volver a la pantalla de niveles. Todos los botones tendrán las mismas propiedades, cambiando el color del boton que variara entre negro, blanco y rojo y el de la letra que sera negro o blanco.

Font Size
 25

Text Color
 #FFFFFF

☒ Visible

Advanced properties

☒ Border Shadow

Font Typeface Import
 Race_Sport.ttf

Título Properties

Common properties

Background Color
 #000000FF

Para poder ordenar todos los recuadros con sus imágenes, crearemos una tabla “table arrangement”, le pondremos 3 columnas y 3 filas. La altura y anchura la dejamos en automático.

Tabla Properties

Common properties

Columns
 3

Height
 Automatic

Width
 Automatic

Rows
 3

Los recuadros blancos, los colocaremos en negro y nos ayudara a la hora de clickar encima de ellos, para que nos diga que hay que decir para cada piloto. Dentro de ellos estarán cada una de las imágenes de 9 pilotos escogidos de los 20 que hay, los más conocidos. Será tan fácil como ajustar las imágenes en el recuadro y listo. Depende la imagen es más o menos grande.

Imagen Properties

Common properties

☒ Clickable

Height
100px

Width
100px

Picture
Checo.png

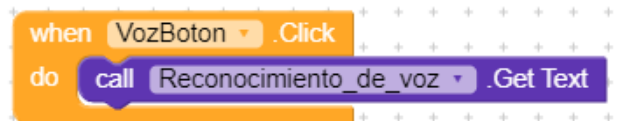
El reconocimiento a voz y texto se configuran en bloques, solo hay que colocarlo.

Blocks (Bloques)

Cuando el usuario hace clic en “VolverNiveles”, nos devuelve a la pantalla llamada “Niveles”.



Cuando el usuario hace clic en “VozBoton”, se llama a “Reconocimiento de voz” para obtener el texto que corresponde a el piloto deseado.



Cuando el usuario hace clic en “Recuadro”, se llama a “Voz” para reproducir el mensaje “nano”. Utilizamos esto para indicarle al usuario que debe decir “nano” en el reconocimiento de voz. De esta manera, sabrá qué palabra debe pronunciar para interactuar con la aplicación, con los demás pilotos igual.



Una vez que “Reconocimiento de voz” obtiene el texto, se establece el idioma a “ES” (español). Luego se comprueba si el resultado del texto obtenido (en minúsculas) es igual a “nano”. Si es así, se reproduce un sonido o archivo de audio llamado “Fernando”.

