

2016

Desarrollo de aplicaciones para Android usando MIT App Inventor 2

Rafael Alberto

Moreno Parra

Contenido

Otros libros del autor	4
Página Web del autor.....	4
Canal de Youtube.....	4
Licencia de este libro	5
Licencia del software	5
Marcas registradas	5
Introducción	6
Paso 1. Tener una cuenta en Gmail	7
Paso 2. Ir a la página de MIT App Inventor 2	7
Paso 3. Preparando el PC para hacer uso de MIT App Inventor 2	7
Paso 4. Instalando el emulador en el PC.....	10
Paso 5. Iniciando con MIT App Inventor	13
Paso 6. Iniciando con el tradicional "Hola Mundo"	15
Paso 7. Leer un par de números y luego sumarlos.....	45
Paso 8. Haciendo operaciones matemáticas más complejas	52
Paso 9. El SI condicional	56
Uso del Y (And) y el O (Or).....	60
Paso 10. Ciclos	63
Paso 11. Operaciones con cadenas (Básico).....	65
A. Longitud de cadenas.....	65
B. Chequear si una cadena está vacía.....	66
C. Quitar espacios al inicio y al final	66
D. Convertir a mayúsculas.....	67
E. Convertir a minúsculas	67
F. Reemplazar una subcadena por otra	68
H. Concatenación de cadenas.....	68
I. Comparación de cadenas.....	70
J. Señala en qué posición se encuentra una determinada subcadena en la cadena	71
K. Retorna verdadero si encuentra la subcadena dentro de la cadena	71
L. Extraer una subcadena	72
Paso 12. Operaciones con listas	73
A. Longitud de una lista	74
B. Seleccionar un ítem de una lista	74
C. Mostrar una lista en un "campo de texto multi-línea"	75
D. Reemplazar un ítem de una lista.....	77
E. Verifica si hay un ítem en una lista.....	77
F. Retorna en qué posición se encuentra un determinado ítem en la lista.....	78
G. Retorna si la lista está llena o vacía.....	79
H. Inserta un ítem en determinada posición de la lista	80
I. Quita un elemento de la lista.....	80
J. Copia una lista en otra	81
K. Adiciona una lista a otra lista.....	81
L. ¿Es determinado objeto una lista?.....	82
Paso 13. Procedimientos y Funciones.....	83
Paso 14. Gráficos	84
A. Dibujar puntos, líneas y círculos.....	85
B. Dibujar textos.....	85
C. Dibujar texto girado en un determinado ángulo	85
D. Tamaño y color del texto	86
E. Fondo del lienzo en un color determinado	86
F. Dibujando una línea con determinado ancho	86
G. Imagen de fondo	87

H. Dibujar círculos con clic de ratón (inicio de captura de eventos).....	90
I. Hacer un sencillo programa de dibujo (captura de eventos)	90
Paso 15. Animación	91
A. Usando un componente de tiempo	91
B. Hacer que una pelota rebote en los bordes (sin requerir componente de tiempo)	93
Paso 16. Bases de datos	94
Paso 17. La capacidad inalámbrica de MIT App Inventor	96
Paso 18. Tomando vídeos.....	100
Paso 19. Tomando fotos.....	100
Paso 20. Ver galería de imágenes.....	101
Paso 21. Usando el acelerómetro	101
Paso 22. Organización en pantalla	102

Otros libros del autor

Libro: "Segunda parte de uso de algoritmos genéticos para la búsqueda de patrones". En Colombia 2014. En publicación por la Universidad Libre – Cali.

Libro: "Un uso de algoritmos genéticos para la búsqueda de patrones". En Colombia 2013. En publicación por la Universidad Libre – Cali.

Libro: "Desarrollo de un evaluador de expresiones algebraicas. **Versión 2.0.** C++, C#, Visual Basic .NET, Java, PHP, JavaScript y Object Pascal (Delphi)". En: Colombia 2013. Págs. 308. Ubicado en: <https://openlibra.com/es/book/evaluador-de-expresiones-algebraicas-ii>

Libro: "Desarrollo de un evaluador de expresiones algebraicas. C++, C#, Visual Basic .NET, Java, PHP, JavaScript y Object Pascal (Delphi)". En: Colombia 2012. Págs. 308. Ubicado en: <https://openlibra.com/es/book/evaluador-de-expresiones-algebraicas>

Libro: "Simulación: Conceptos y Programación" En: Colombia 2012. Págs. 81. Ubicado en: <https://openlibra.com/es/book/simulacion-conceptos-y-programacion>

Libro: "Desarrollo de videojuegos en 2D con Java y Microsoft XNA". En: Colombia 2011. Págs. 260. Ubicado en: <https://openlibra.com/es/book/desarrollo-de-juegos-en-2d-usando-java-y-microsoft-xna> . ISBN: 978-958-8630-45-8

Libro: "Desarrollo de gráficos para PC, Web y dispositivos móviles" En: Colombia 2009. ed.: Artes Gráficas Del Valle Editores Impresores Ltda. ISBN: 978-958-8308-95-1 v. 1 págs. 317

Artículo: "Programación Genética: La regresión simbólica".
Entramado ISSN: 1900-3803 ed.: Universidad Libre Seccional Cali
v.3 fasc.1 p.76 - 85, 2007

Página Web del autor

Investigación sobre Inteligencia Artificial: <http://darwin.50webs.com>

Canal de Youtube

Canal en Youtube: <http://www.youtube.com/user/RafaelMorenoP> (dedicado a desarrollo de aplicaciones web en JavaScript y PHP, uso de aplicaciones Web escritas en PHP, desarrollo en C# y Visual Basic .NET)

Licencia de este libro



Licencia del software

Todo el software desarrollado aquí tiene licencia LGPL "Lesser General Public License"



Marcas registradas

En este libro se hace uso de las siguientes tecnologías registradas:

MIT® AppInventor 2® Enlace: <http://appinventor.mit.edu/explore/front.html>

Microsoft® Windows® Enlace: <http://windows.microsoft.com/en-US/windows/home>

Introducción

Abordar un desarrollo en Android con el lenguaje de programación Java, XML, emulación virtual y ambiente de dispositivo móvil puede ser intimidante. Esa es la razón por la que Google ideó una forma sencilla de desarrollar aplicaciones para Android: haciendo uso de programación gráfica y atraer así a los desarrolladores. Google, sin embargo, dejó de lado este proyecto el cual fue retomado por el MIT y se ha mantenido así.

El libro sigue el formato de llevar paso a paso al lector desde preparar el ambiente de desarrollo, pasando por el diseño de la pantalla en un Smartphone Android, los diferentes componentes visuales y no visuales, programación usando bloques como si fuese un rompecabezas, variables, uso del sí condicional, ciclos o bucles, hasta el uso de las capacidades propias de un Smartphone como la cámara, video o el acelerómetro.

MIT App Inventor 2 oficialmente es una herramienta para desarrollo en línea por lo que requiere conexión permanente a Internet.

Este libro es una actualización de un libro anterior que quedó obsoleto cuando el MIT App Inventor actualizó su tecnología.

Paso 1. Tener una cuenta en Gmail

En <https://accounts.google.com/> puede crear una cuenta de correo, ¿ya la tiene? Entonces este paso es innecesario.

Paso 2. Ir a la página de MIT App Inventor 2

Diríjase a <http://appinventor.mit.edu/explore/front.html> . Allí inicia todo, pero antes hay que preparar el PC para poder usar esta herramienta.

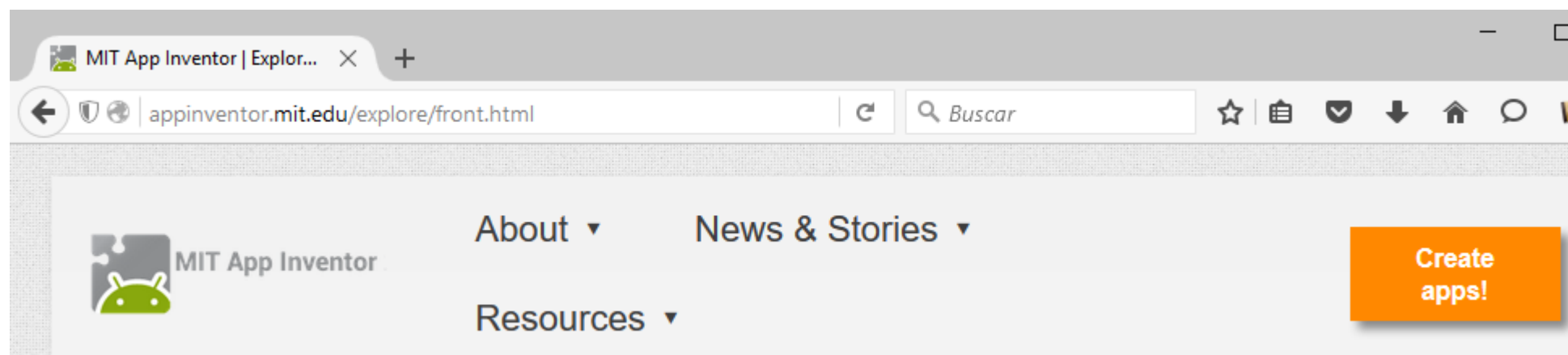


Imagen 1: Página inicial de MIT App Inventor 2

Paso 3. Preparando el PC para hacer uso de MIT App Inventor 2

En la siguiente dirección <http://appinventor.mit.edu/explore/get-started.html> están los pasos a dar para iniciar con esta herramienta

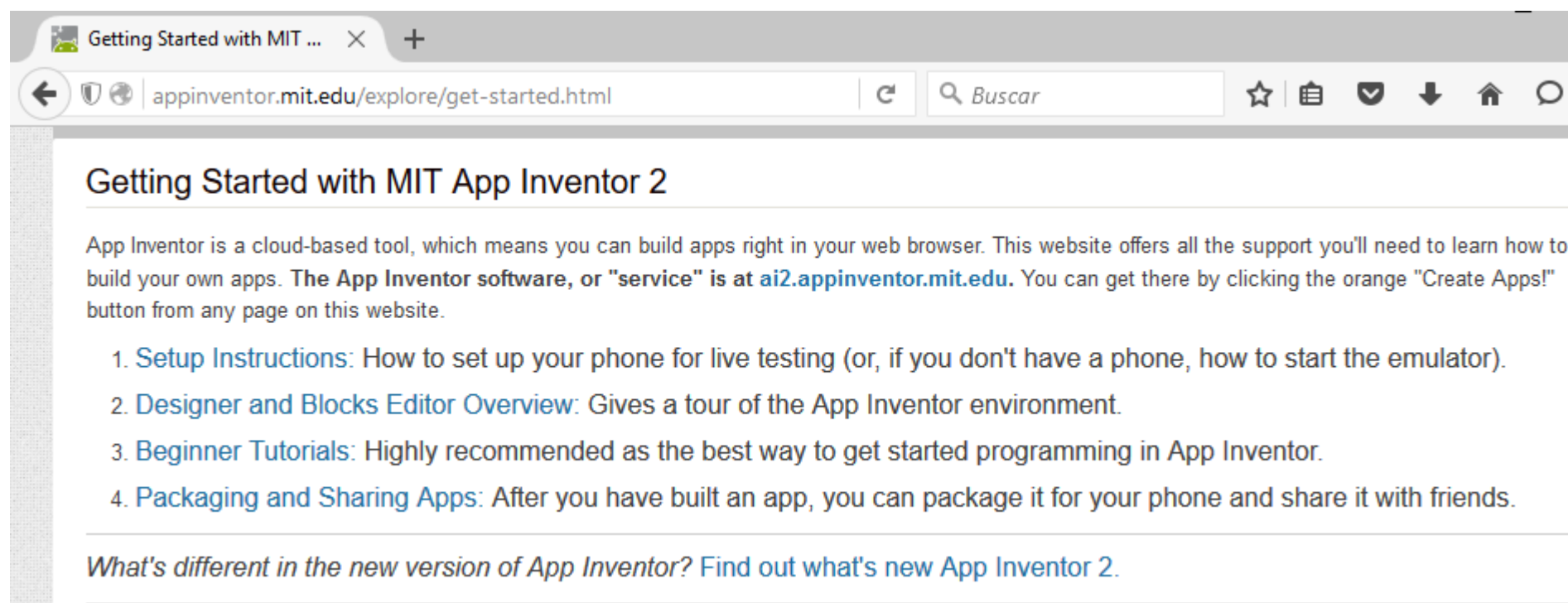


Imagen 2: Preparando el terreno para usar MI App Inventor 2

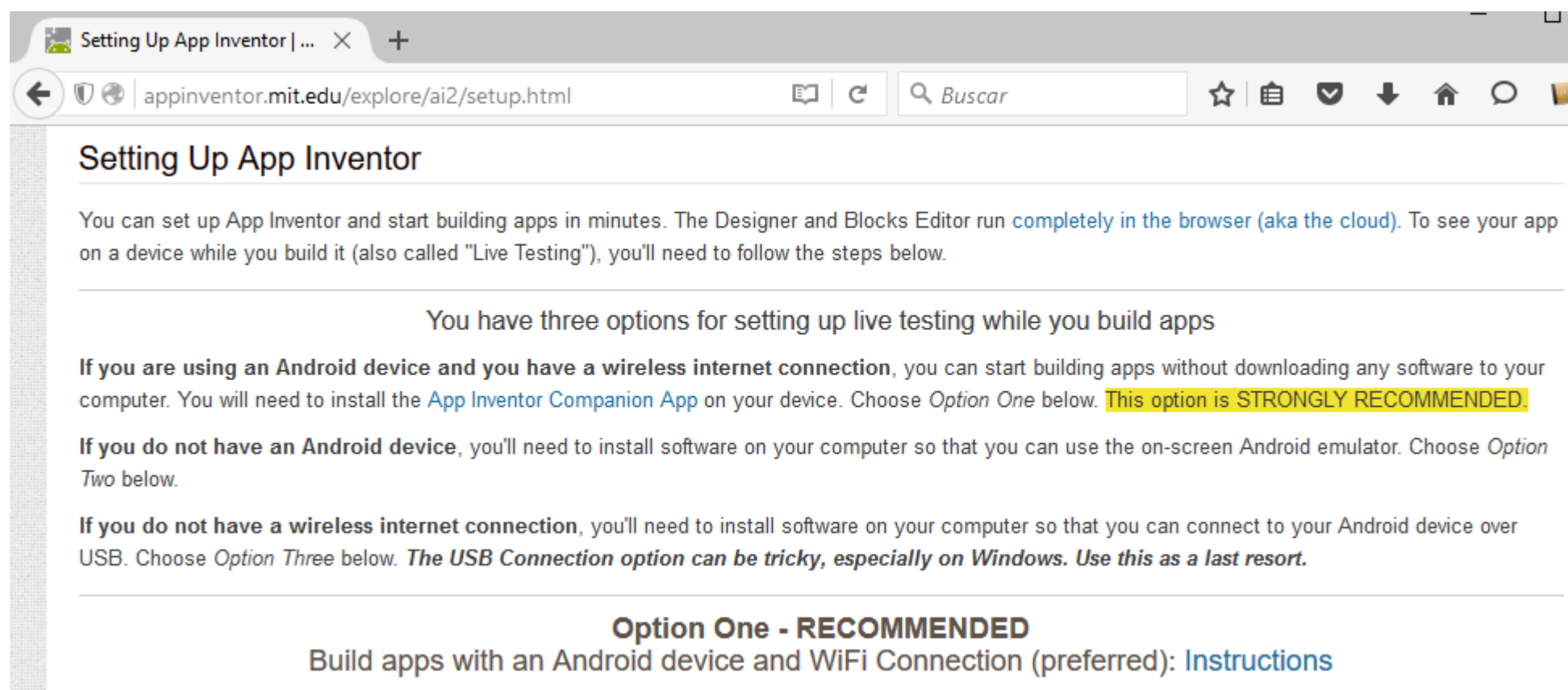


Imagen 3: Hay tres opciones para probar las aplicaciones en Android

En el sitio oficial recomiendan que las aplicaciones se desarrollen en el PC y se prueben vía WiFi en un dispositivo Android.

A falta de un dispositivo Android se opta por instalar un emulador en el PC. En <http://appinventor.mit.edu/explore/ai2/setup-emulator.html> se muestra como instalar ese emulador y para Windows se descarga en http://appinv.us/aisetup_windows

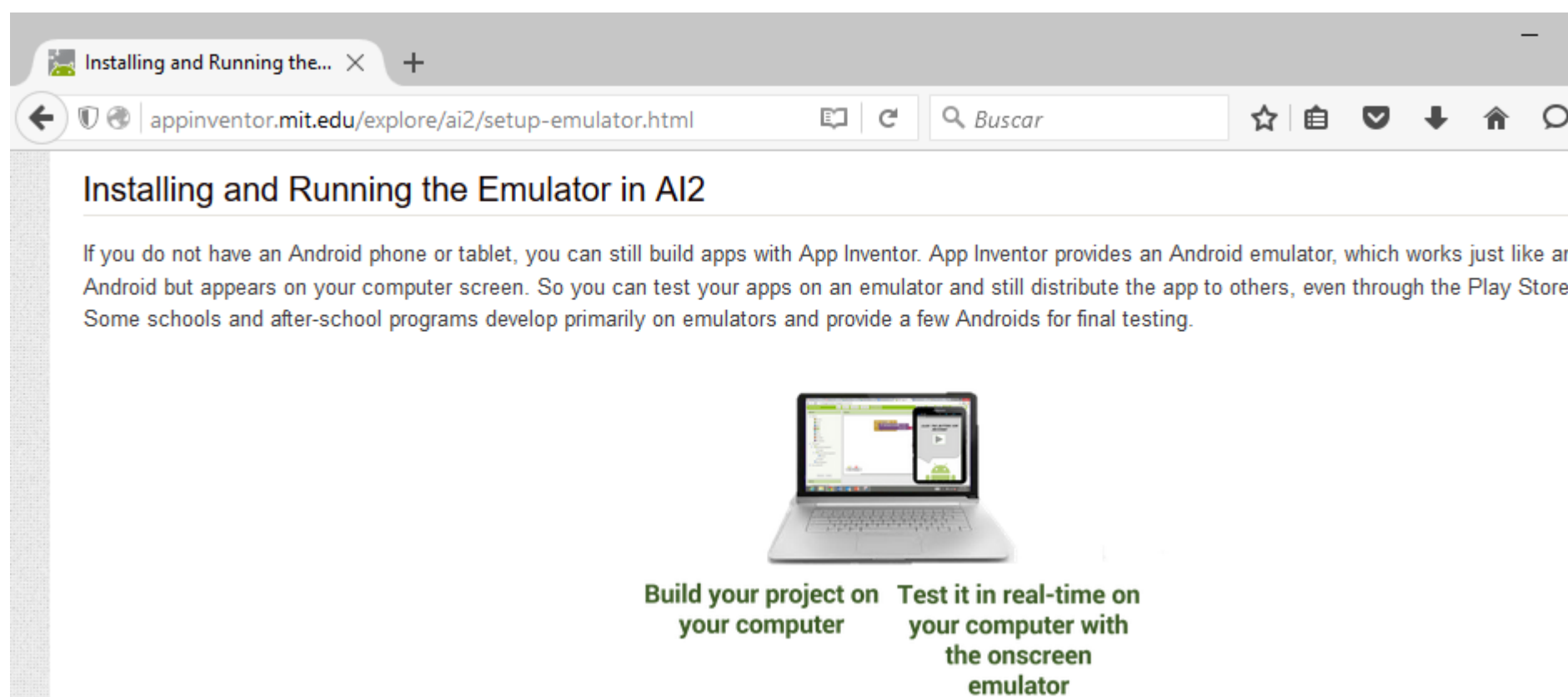


Imagen 4: Instalando el emulador

Este es el instalador

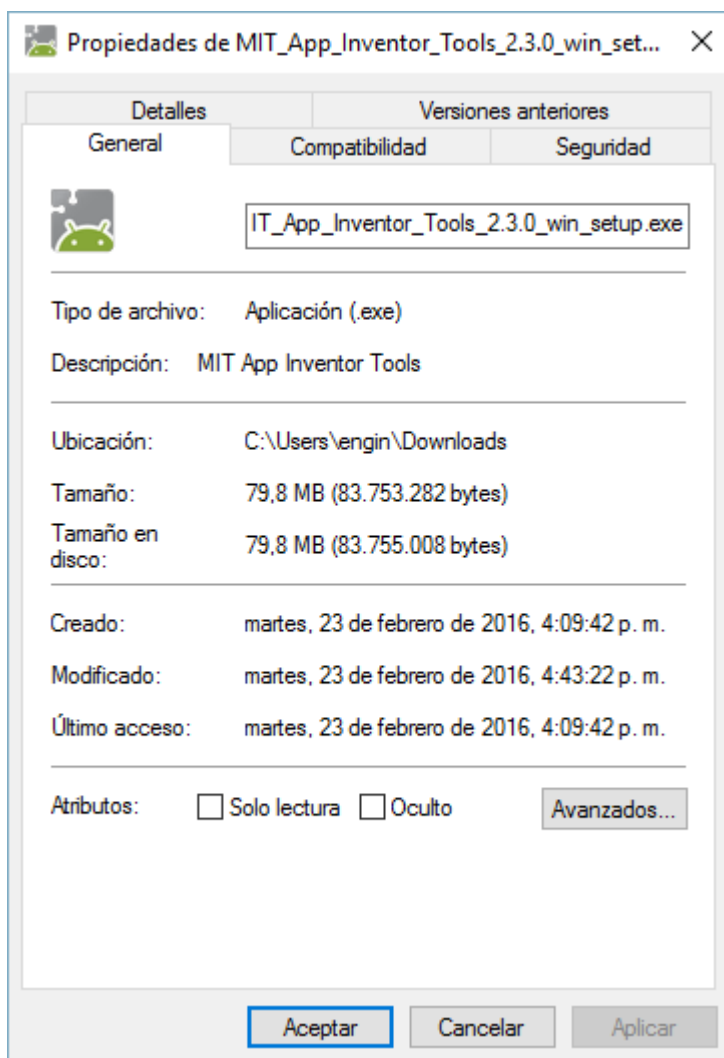


Imagen 5: Propiedades del instalador del emulador

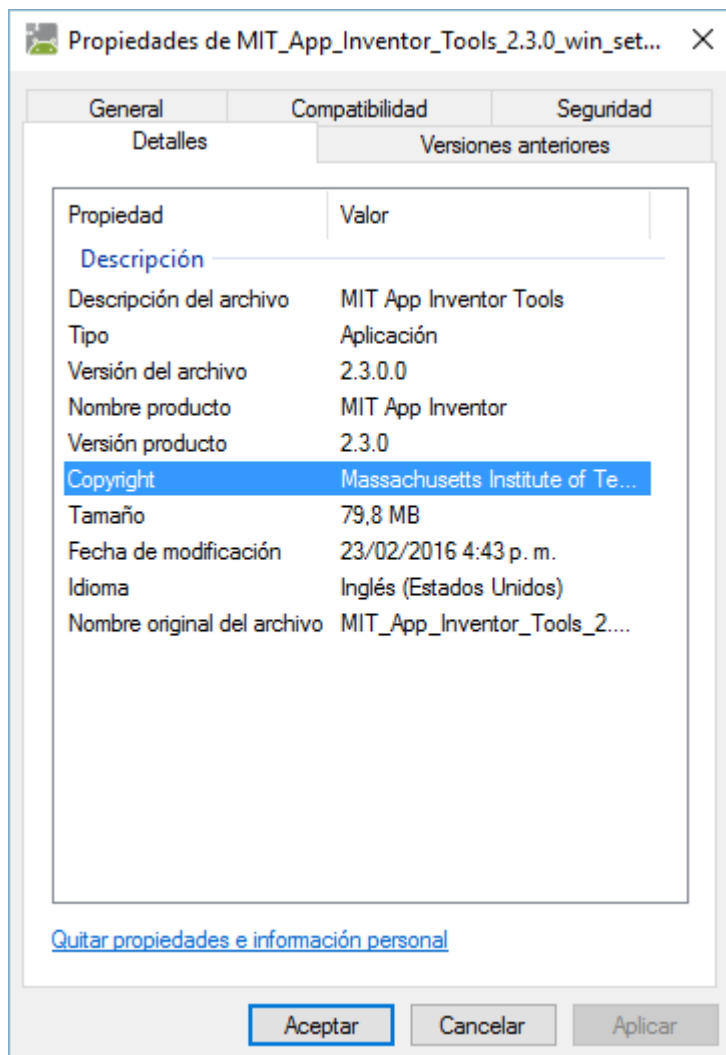


Imagen 6: Detalles del instalador del emulador

Paso 4. Instalando el emulador en el PC

Descargado el instalador del emulador se procede a ejecutarlo



Imagen 7: Inicio de la instalación del emulador

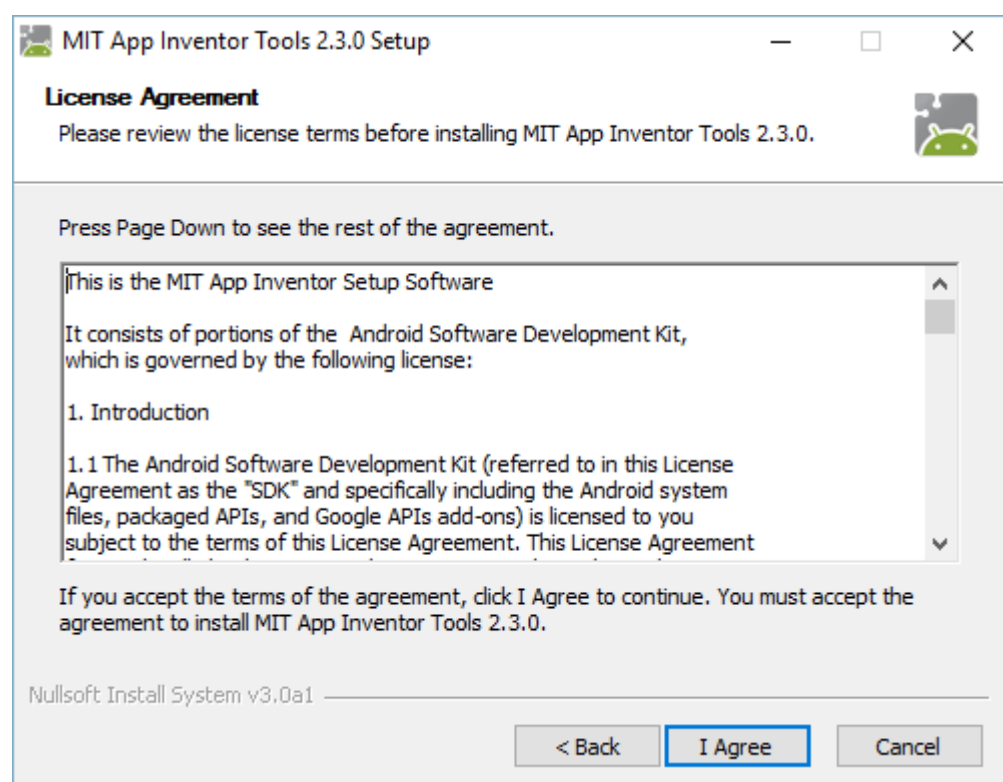


Imagen 8: Leer y aprobar la licencia

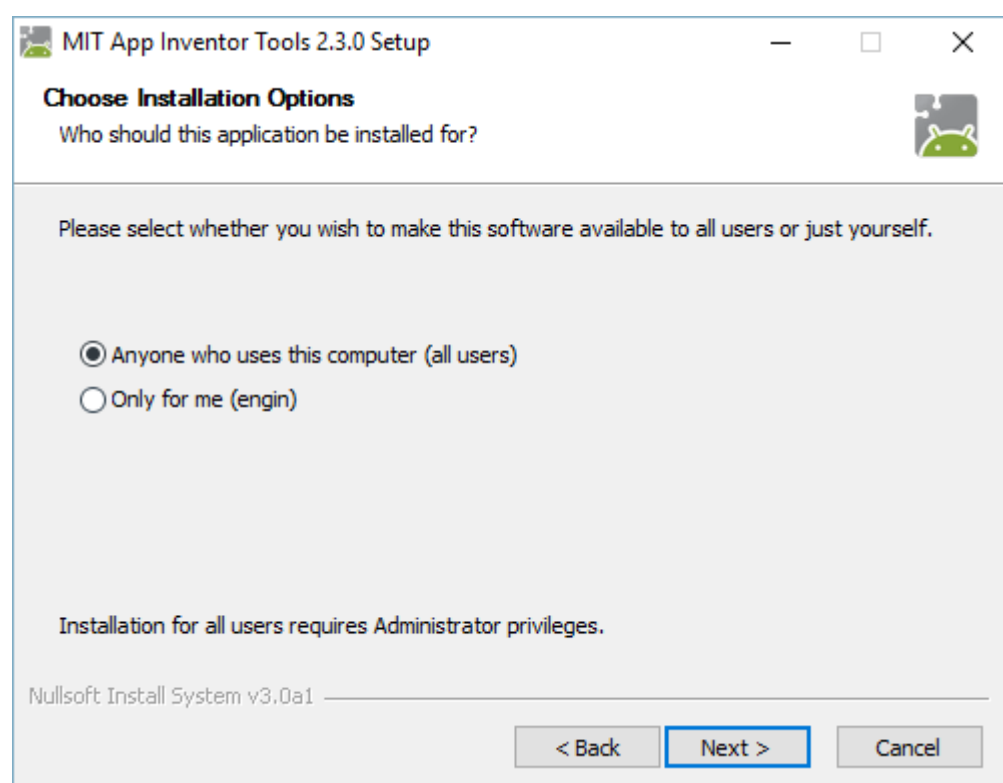


Imagen 9: Instalación para todos los usuarios

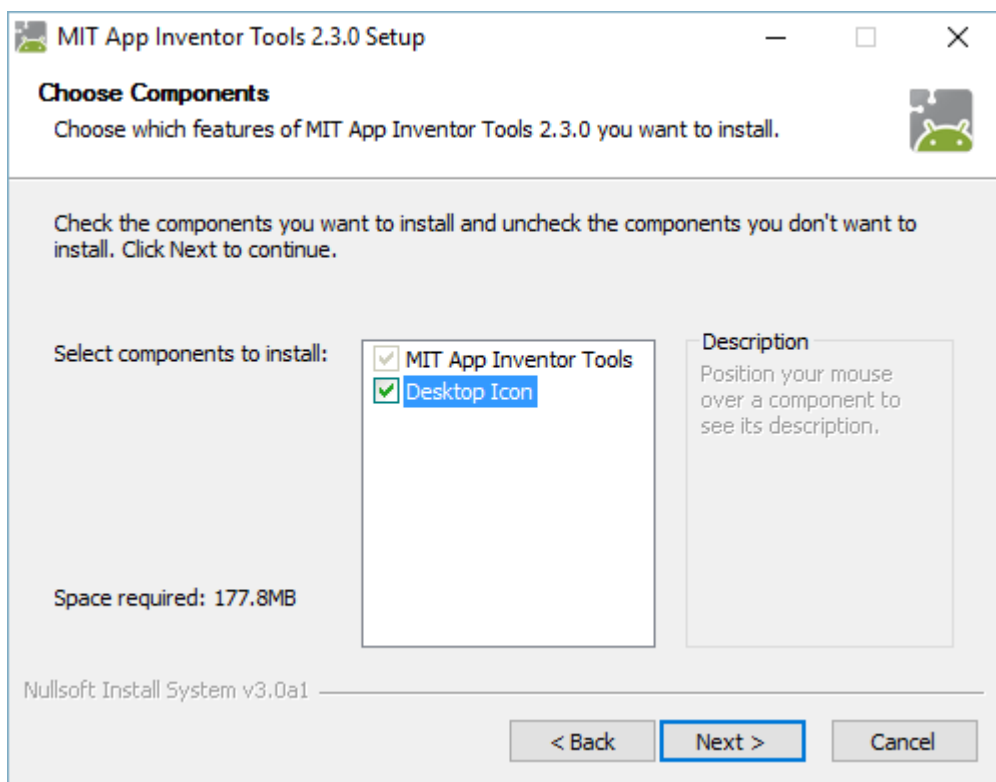


Imagen 10: Se recomienda crear un icono de lanzamiento en el escritorio

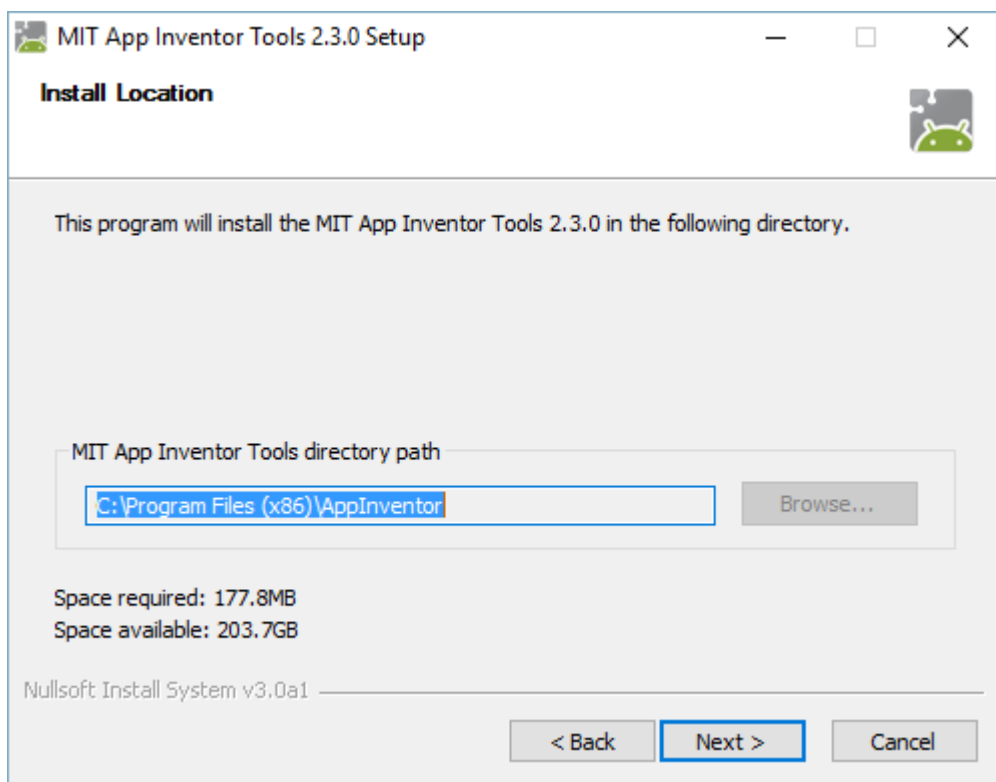


Imagen 11: Ubicación donde se desee instalar el emulador

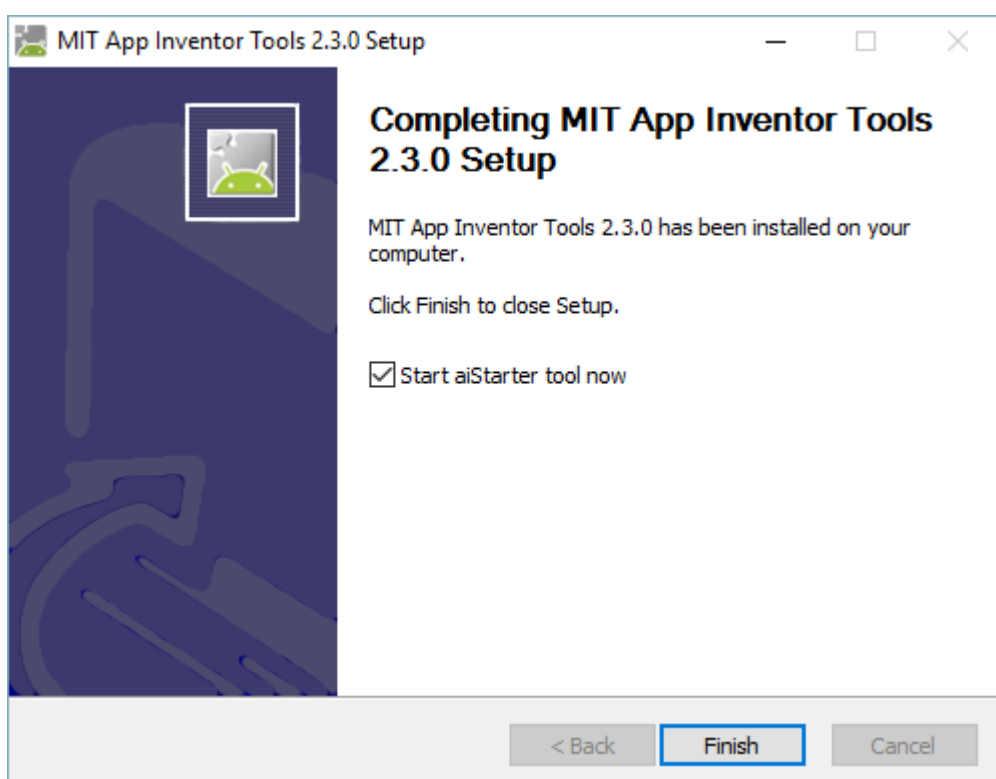
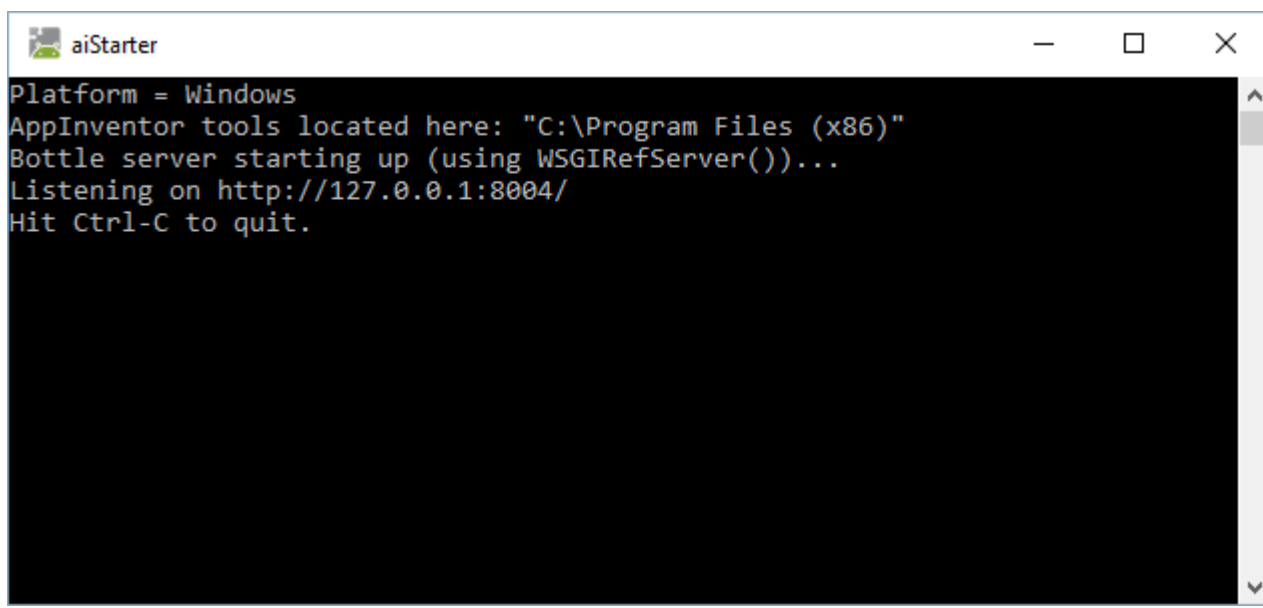


Imagen 12: Finalizada la instalación. Se ejecuta el emulador



```
aiStarter
Platform = Windows
AppInventor tools located here: "C:\Program Files (x86)"
Bottle server starting up (using WSGIRefServer())...
Listening on http://127.0.0.1:8004/
Hit Ctrl-C to quit.
```

Imagen 13: Emulador ejecutándose. No cierre esta ventana.

Paso 5. Iniciando con MIT App Inventor

Ya puede dirigirse a <http://appinventor.mit.edu/explore/index-2.html> y dar clic en "Create Apps"

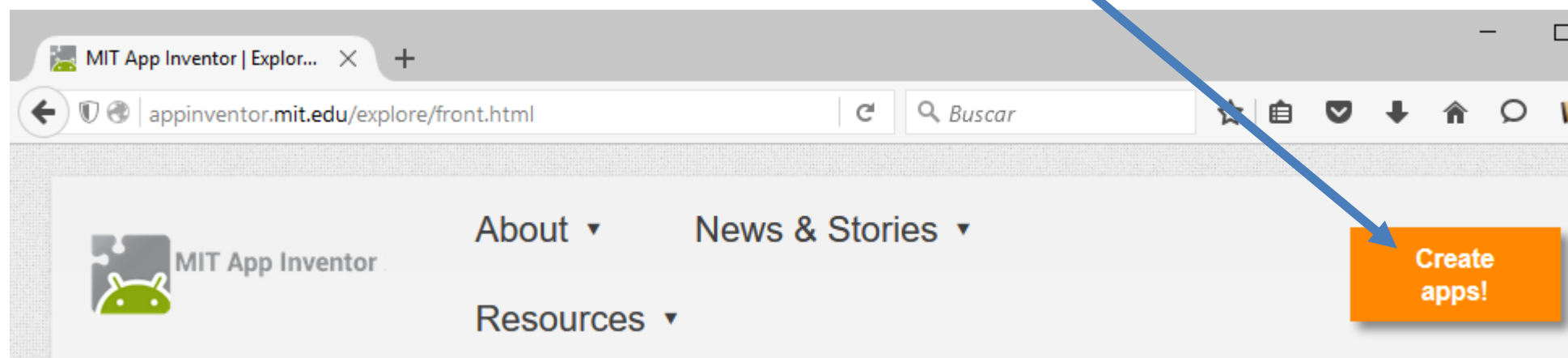


Imagen 14: De clic en "Create Apps" arriba a la derecha

Luego entre con su cuenta de Gmail

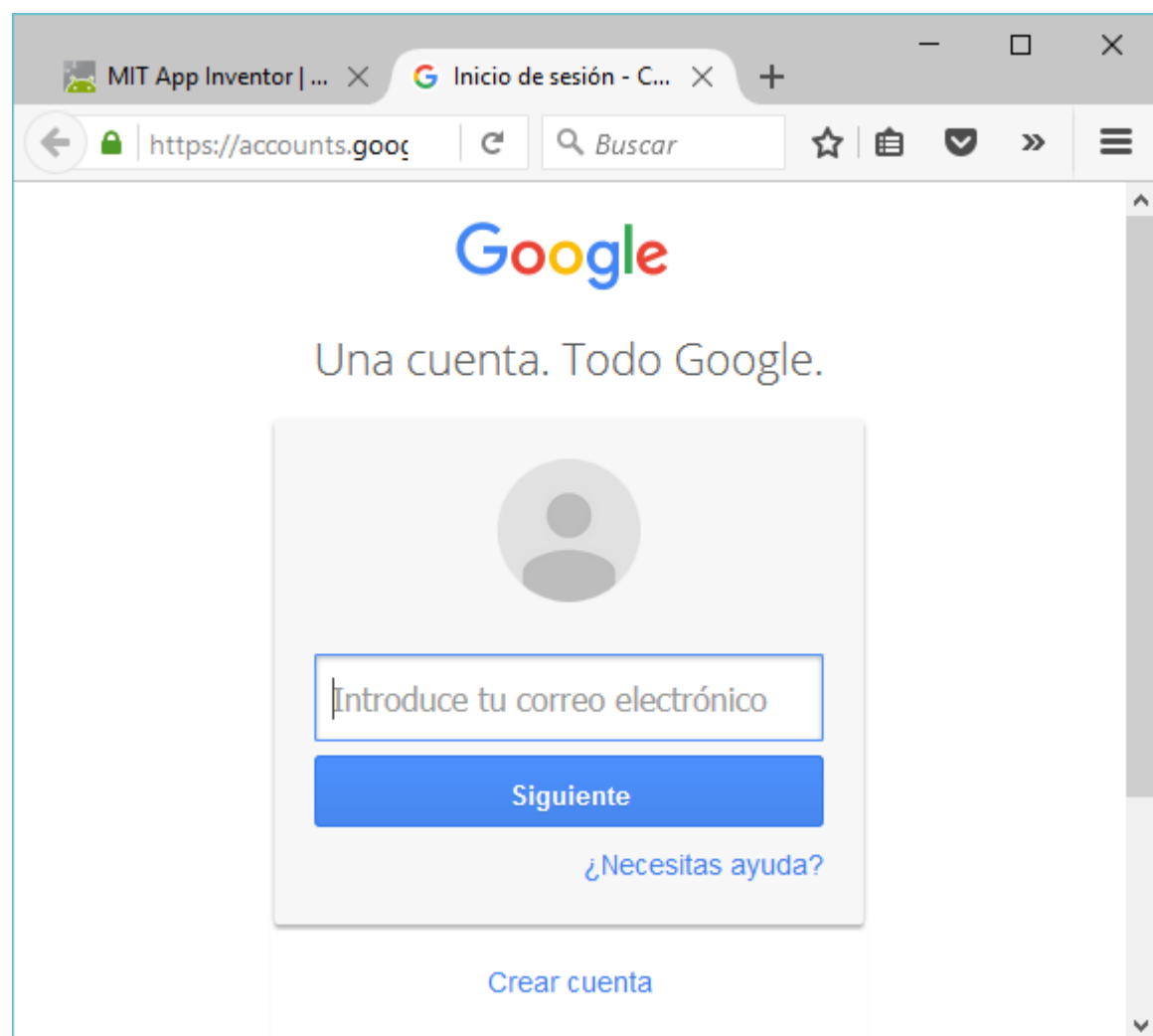


Imagen 15: Entre con su cuenta de Gmail

Esta es la pantalla inicial

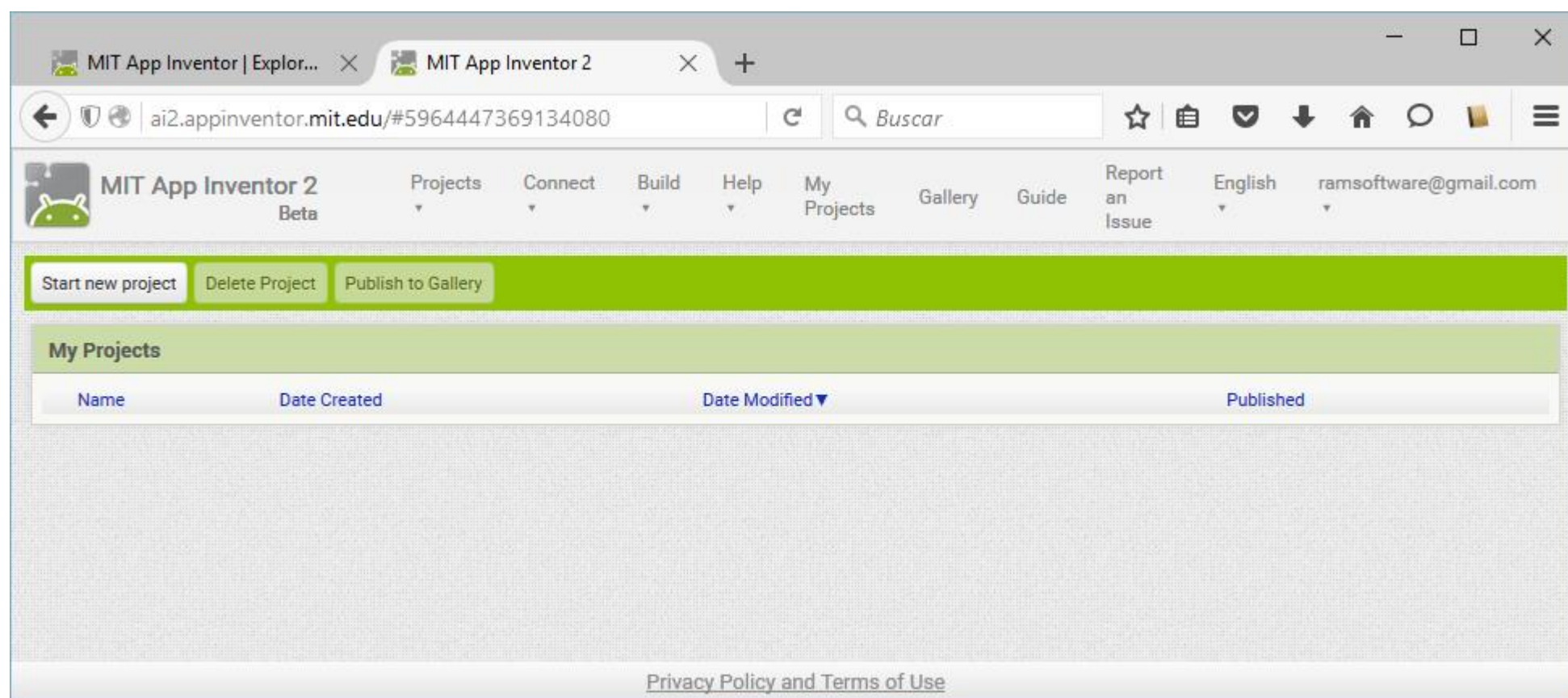


Imagen 16: Pantalla inicial que muestra los proyectos que ha hecho

Como es la primera vez que inicia en esta herramienta, todavía no tiene proyectos.

Paso 6. Iniciando con el tradicional “Hola Mundo”

Puede cambiar el idioma de la herramienta

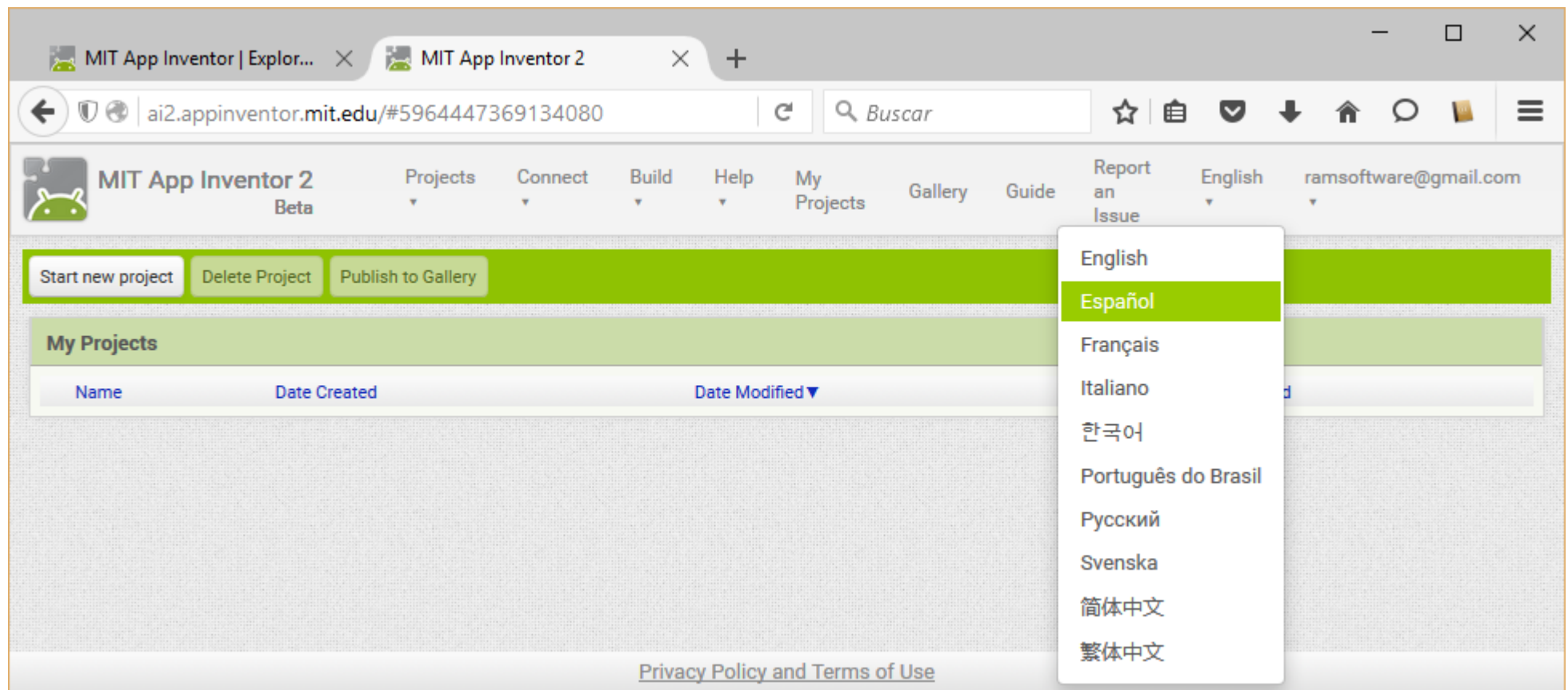


Imagen 17: Puede cambiar el idioma de la nueva herramienta

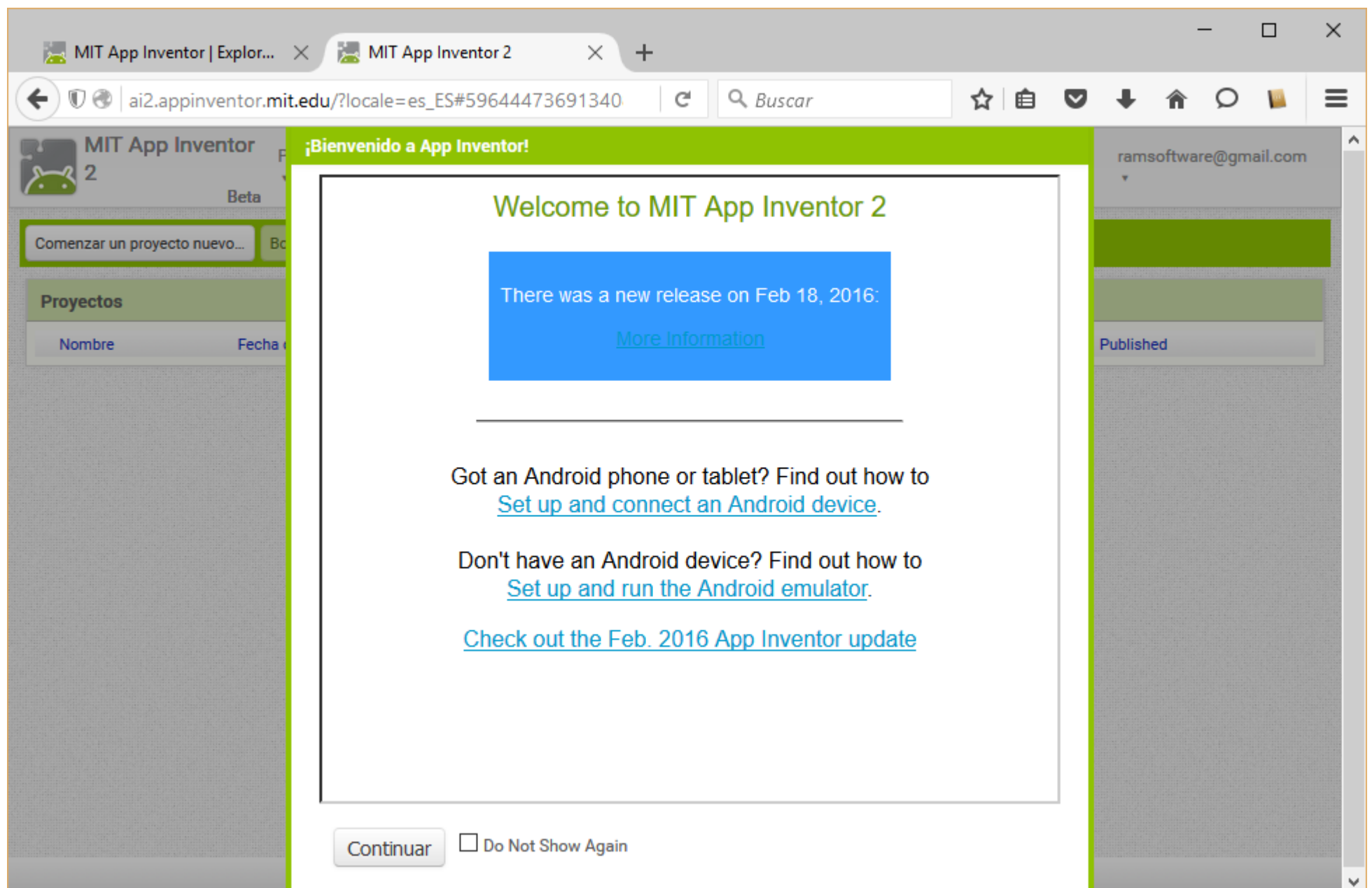


Imagen 18: La interfaz en español

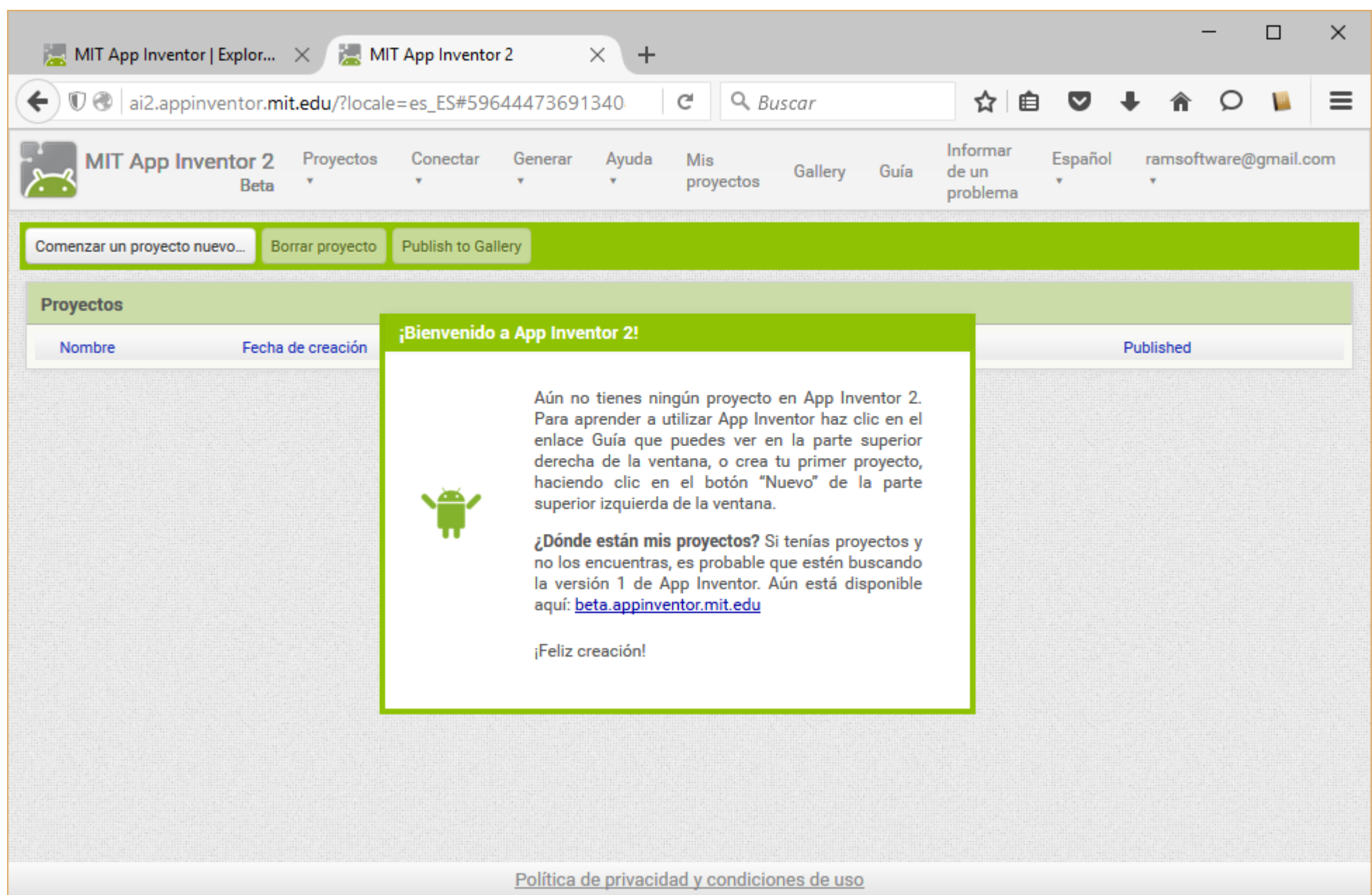


Imagen 19: Bienvenida a la herramienta

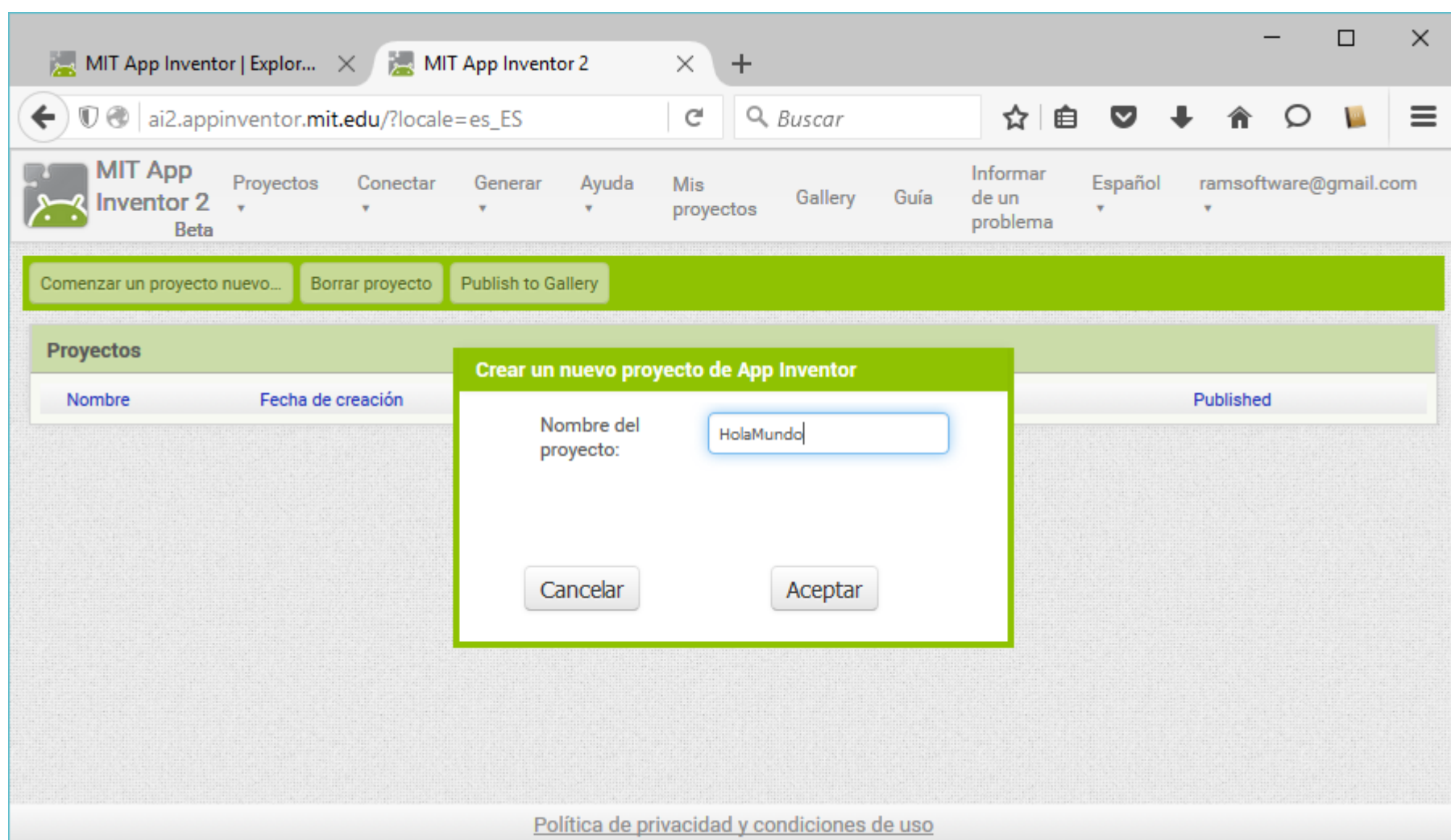


Imagen 20: Se le pone un nombre al proyecto (espacios no están permitidos)

Inmediatamente al crear un proyecto, se muestra la página de diseño de pantallas, la cual es muy parecida a otros IDE (Integrated Development Environment)

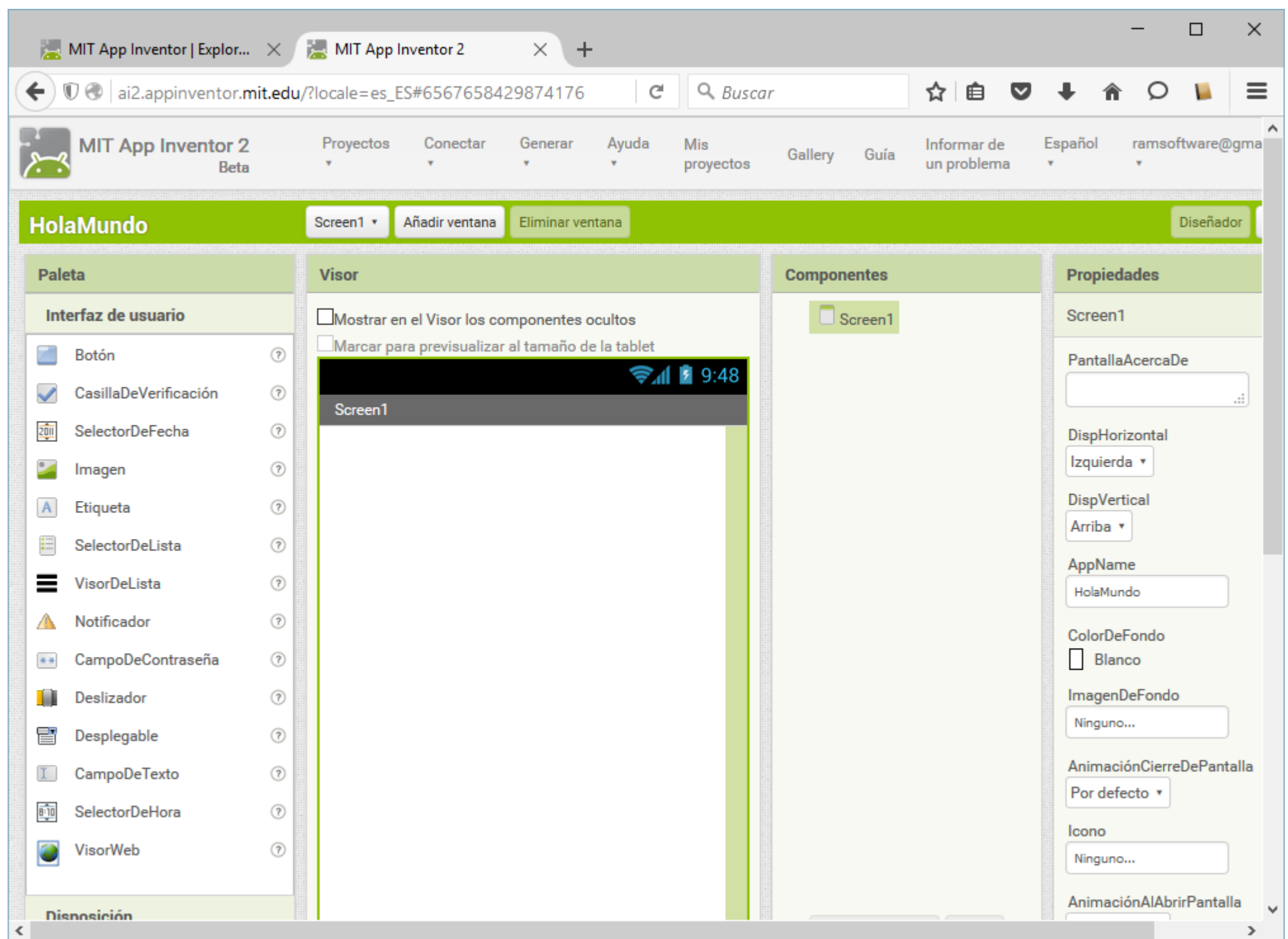


Imagen 21: Pantalla de diseño de pantalla.

Arrastre el control gráfico "CampoDeTexto" a la pantalla "Screen1" y observe lo que sucede

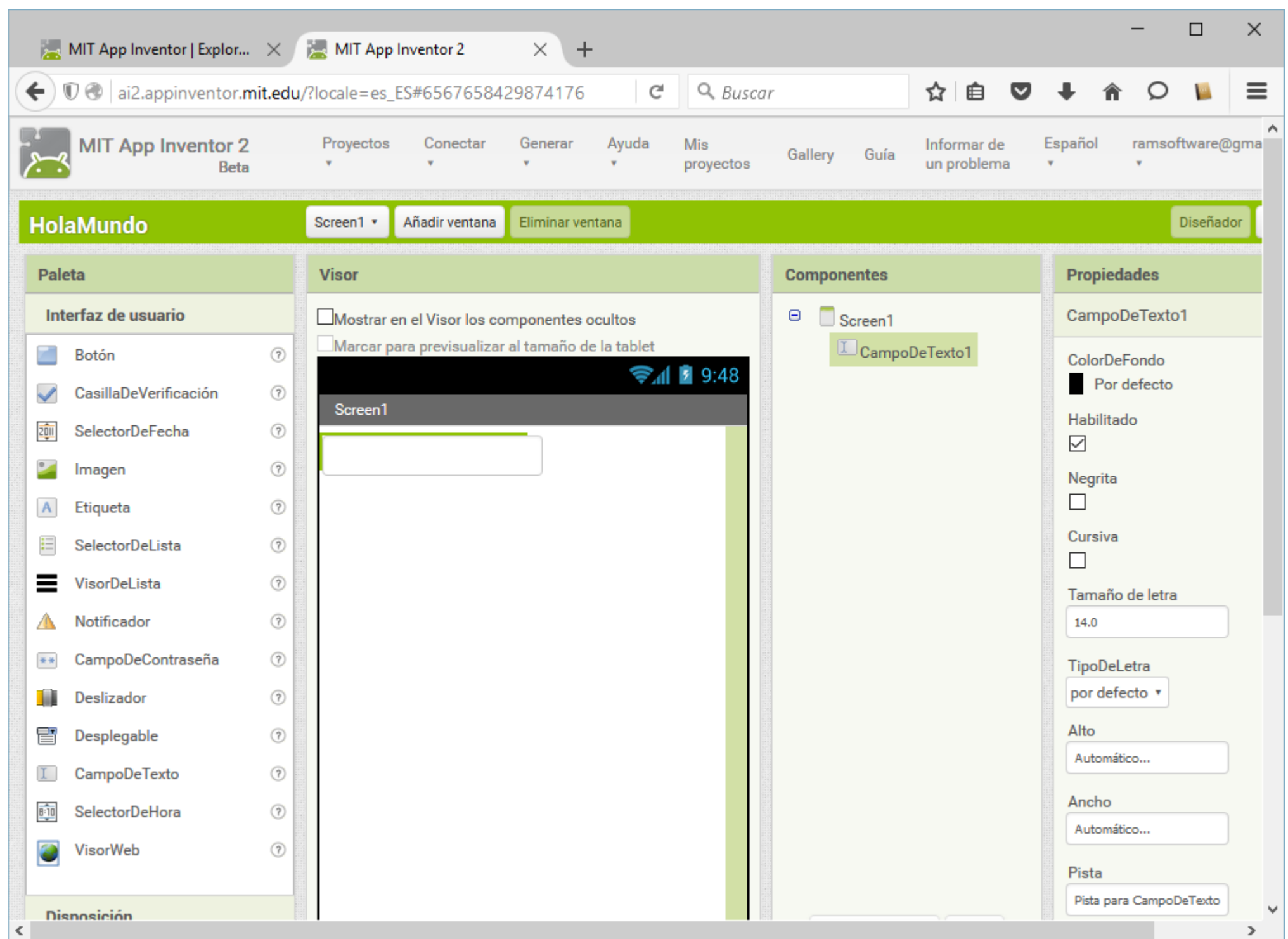


Imagen 22: Arrastre el control gráfico "TextBox" a la pantalla "Screen1"

Arrastre un "Botón" a la pantalla "Screen1"

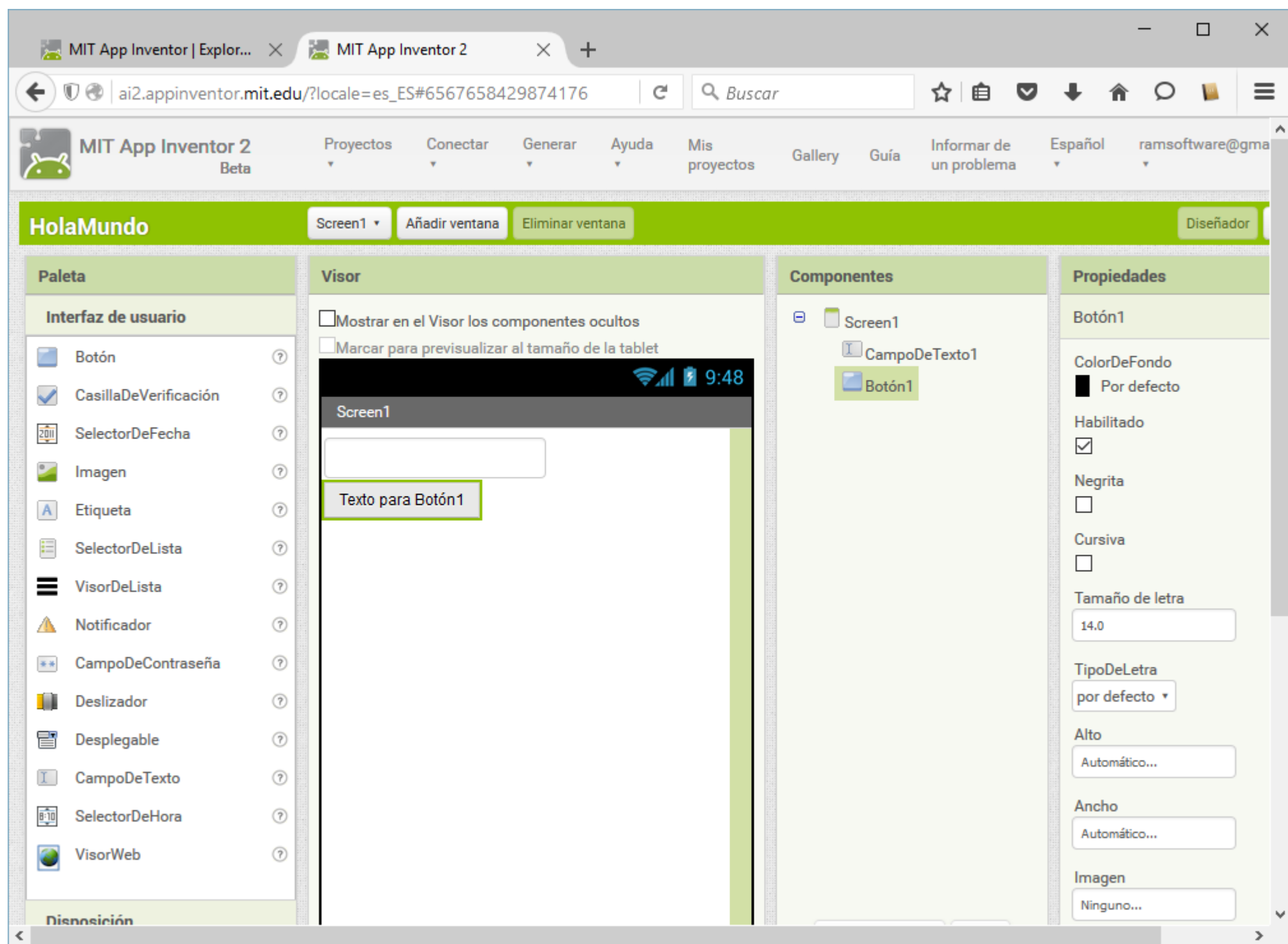


Imagen 23: Arrastre un "Button" a la pantalla "Screen1"

Una muy buena recomendación, es renombrar los objetos gráficos con un estándar. Se usará esta forma:

Objeto que usará	Prefijo y forma de nombrar	Ejemplos
Botón (Button)	btn + Letra mayúscula + resto de nombre	btnAceptar, btnPresionar, btnProcesar
Campo de texto (TextBox)	txt + Letra mayúscula + resto de nombre	txtAviso, txtNombre, txtApellido

Se presiona el botón "Rename" para cambiar cada objeto gráfico.

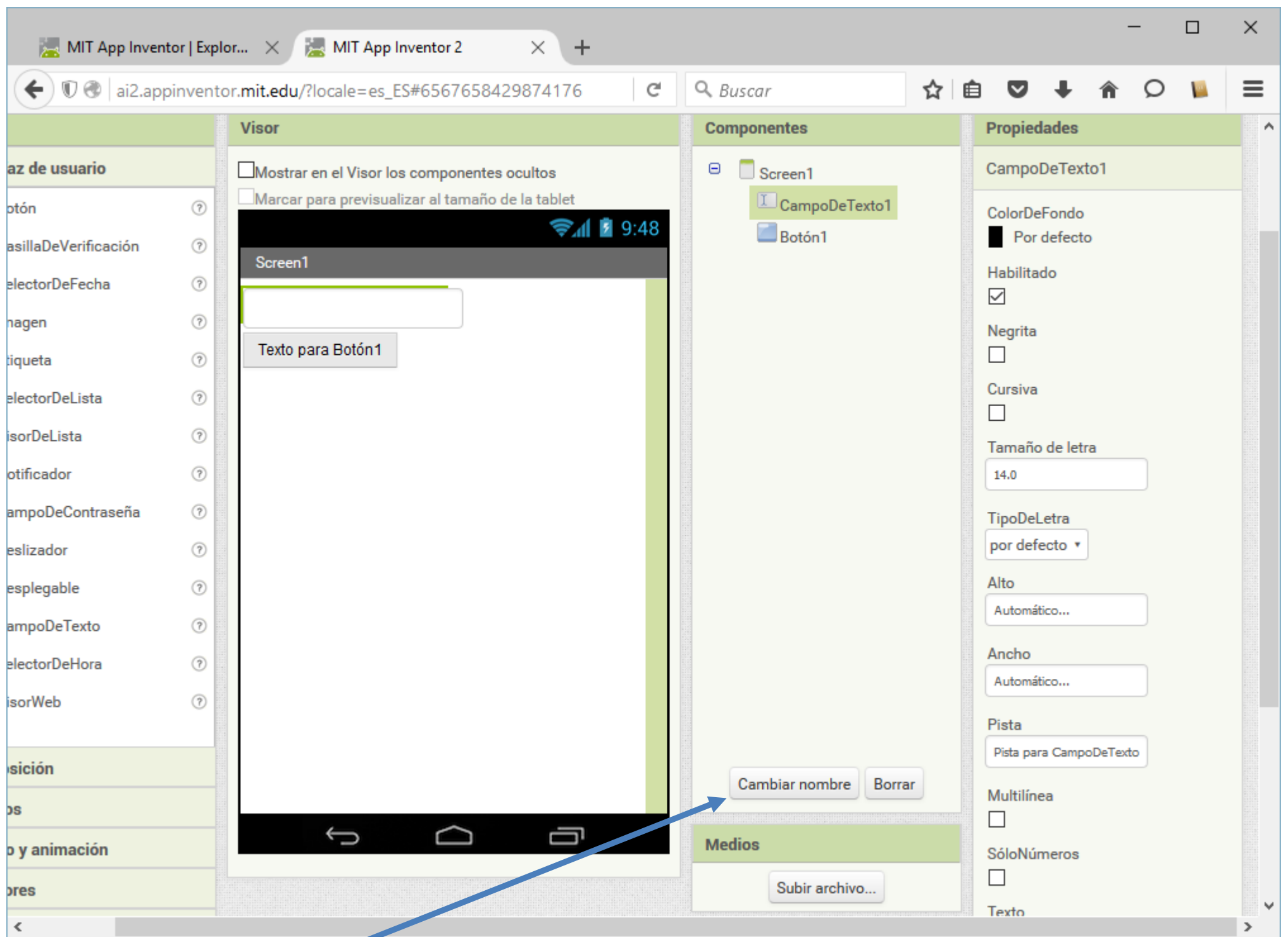


Imagen 24: Renombrar los objetos

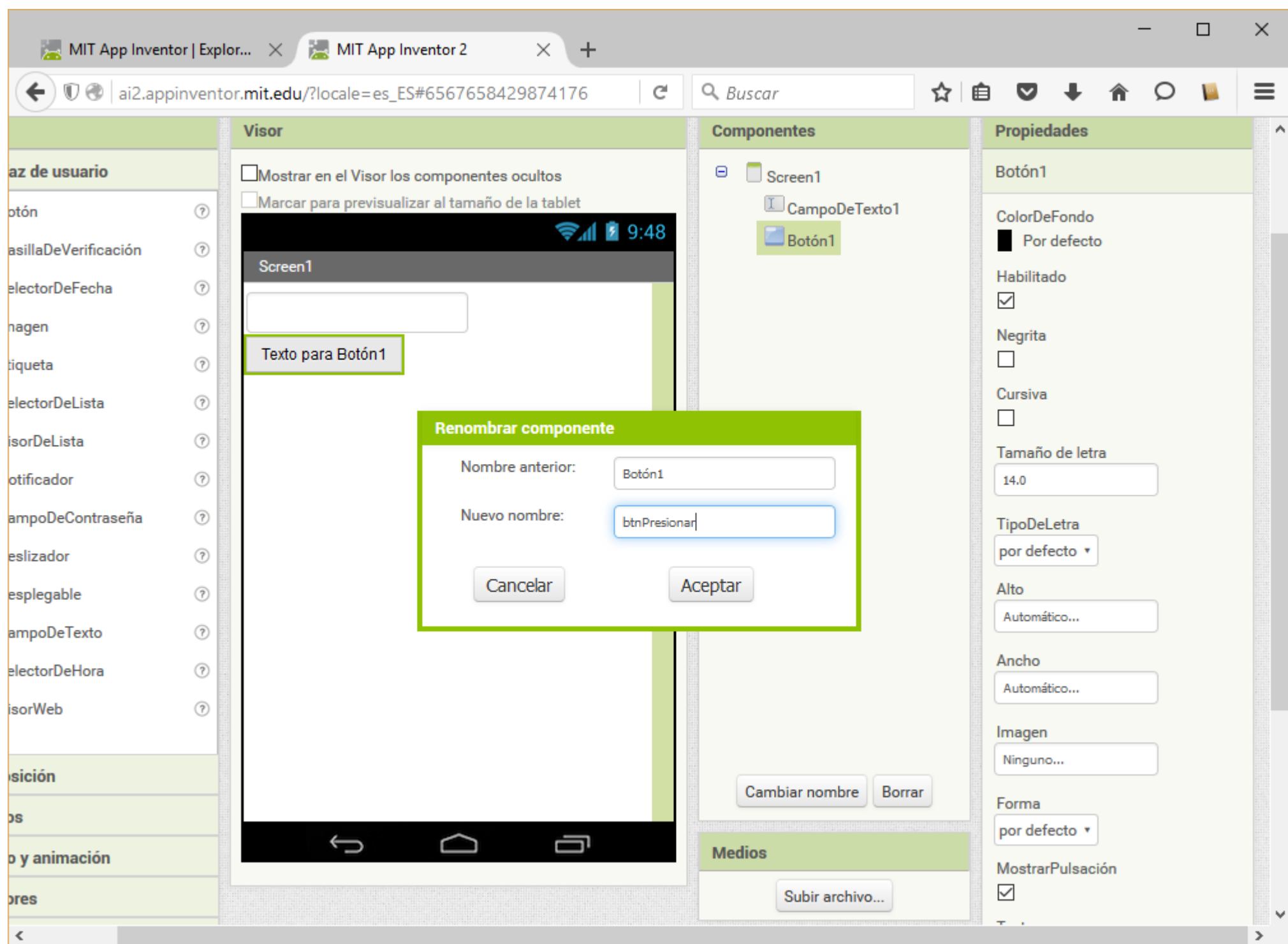


Imagen 25: Renombra el objeto visual "Botón"

El siguiente paso es cambiar las propiedades (que se encuentran a la derecha) de cada objeto visual utilizado hasta lograr el efecto deseado.

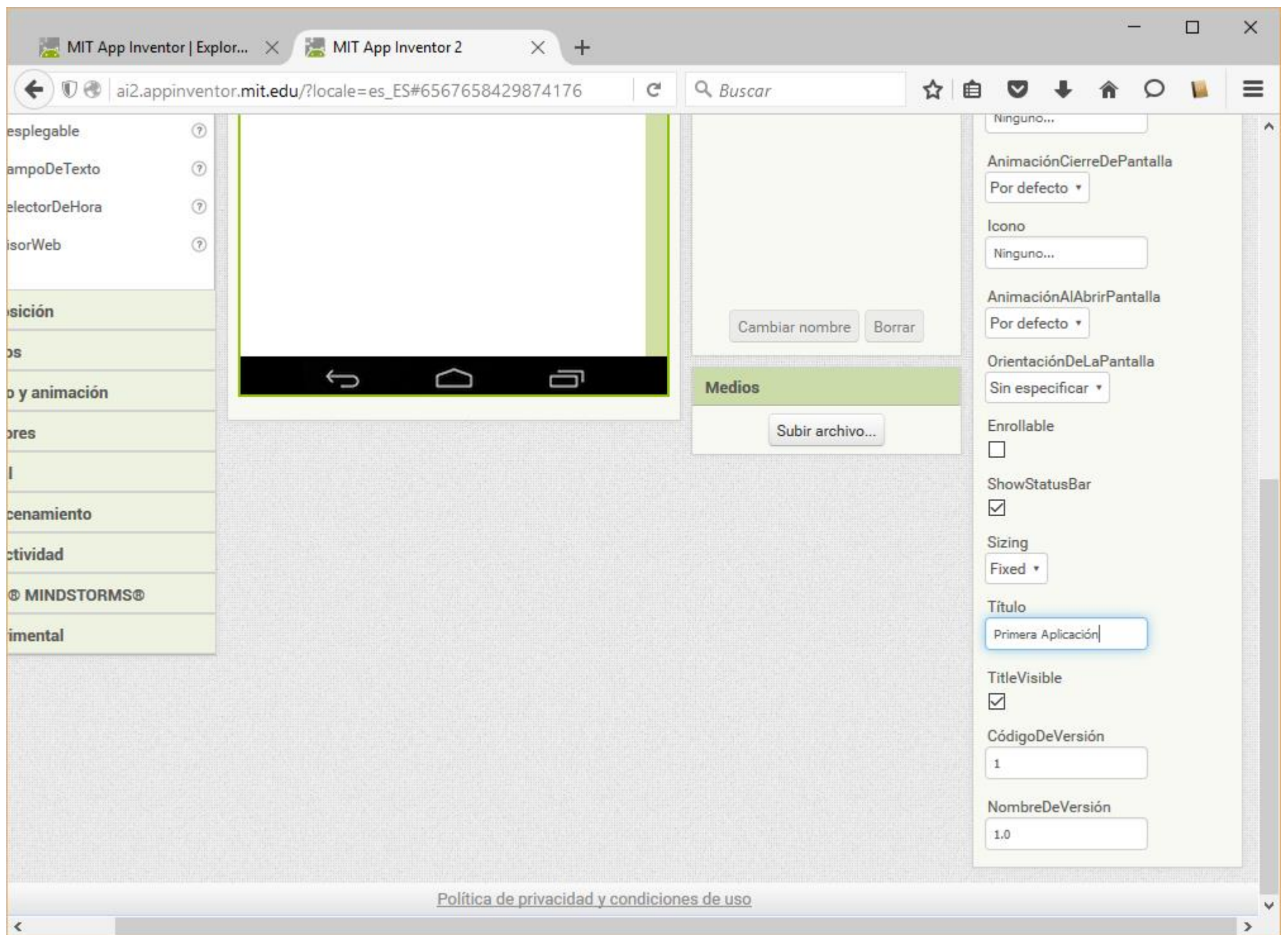


Imagen 26: Cambia la propiedad "título" de la pantalla

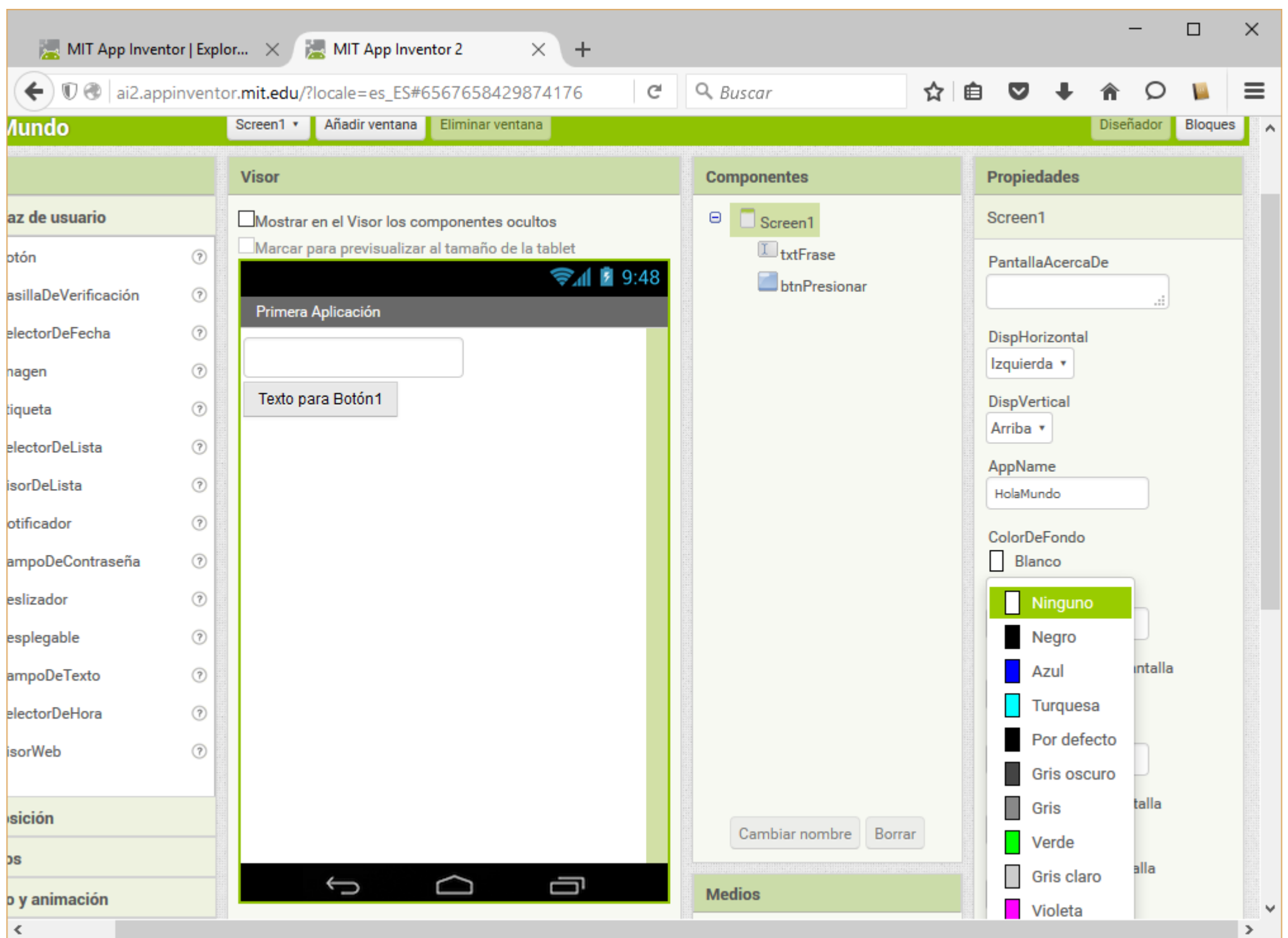


Imagen 27: Cambia el color de fondo de la pantalla

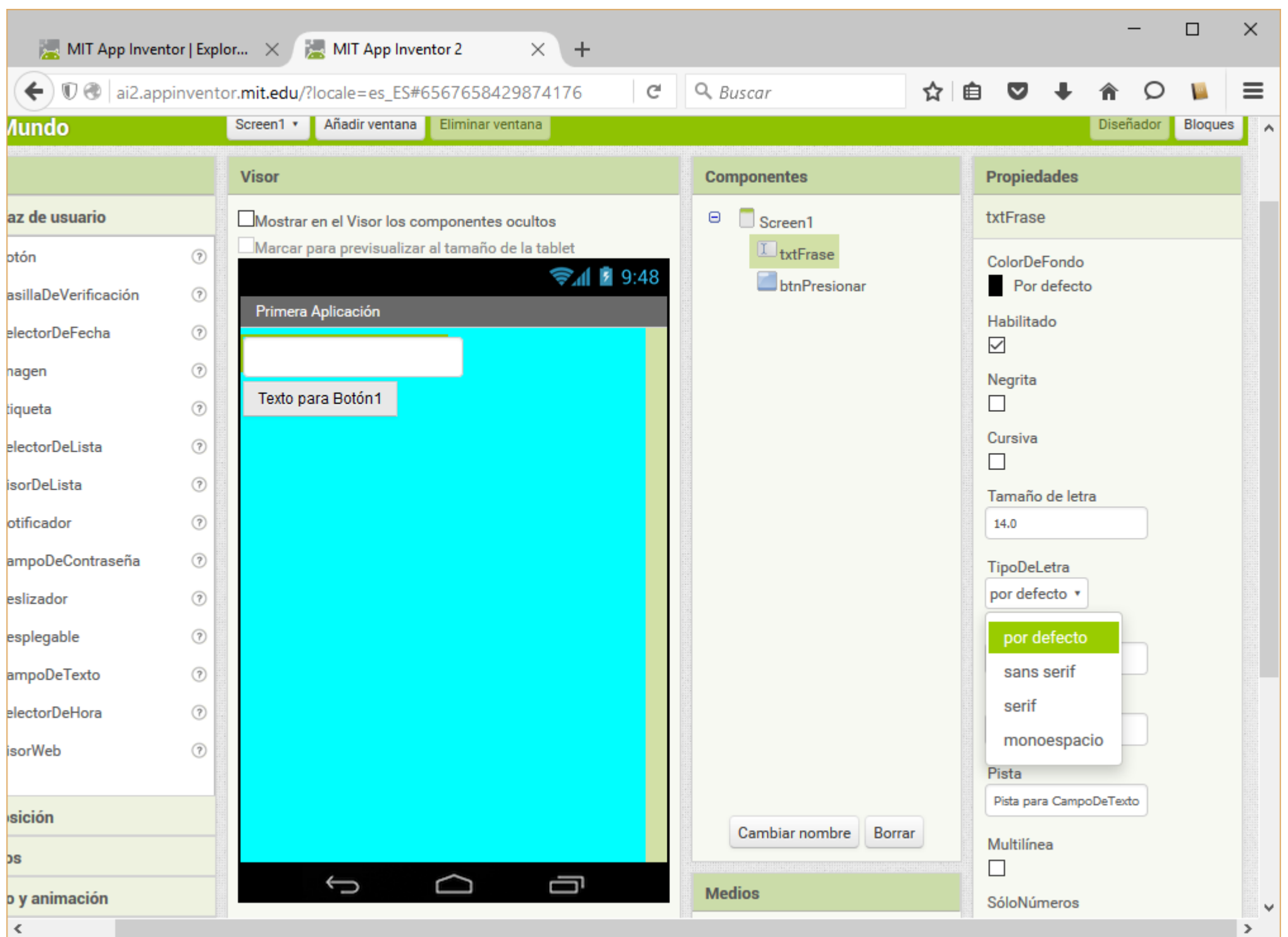


Imagen 28: Cambia las propiedades del Campo de Texto

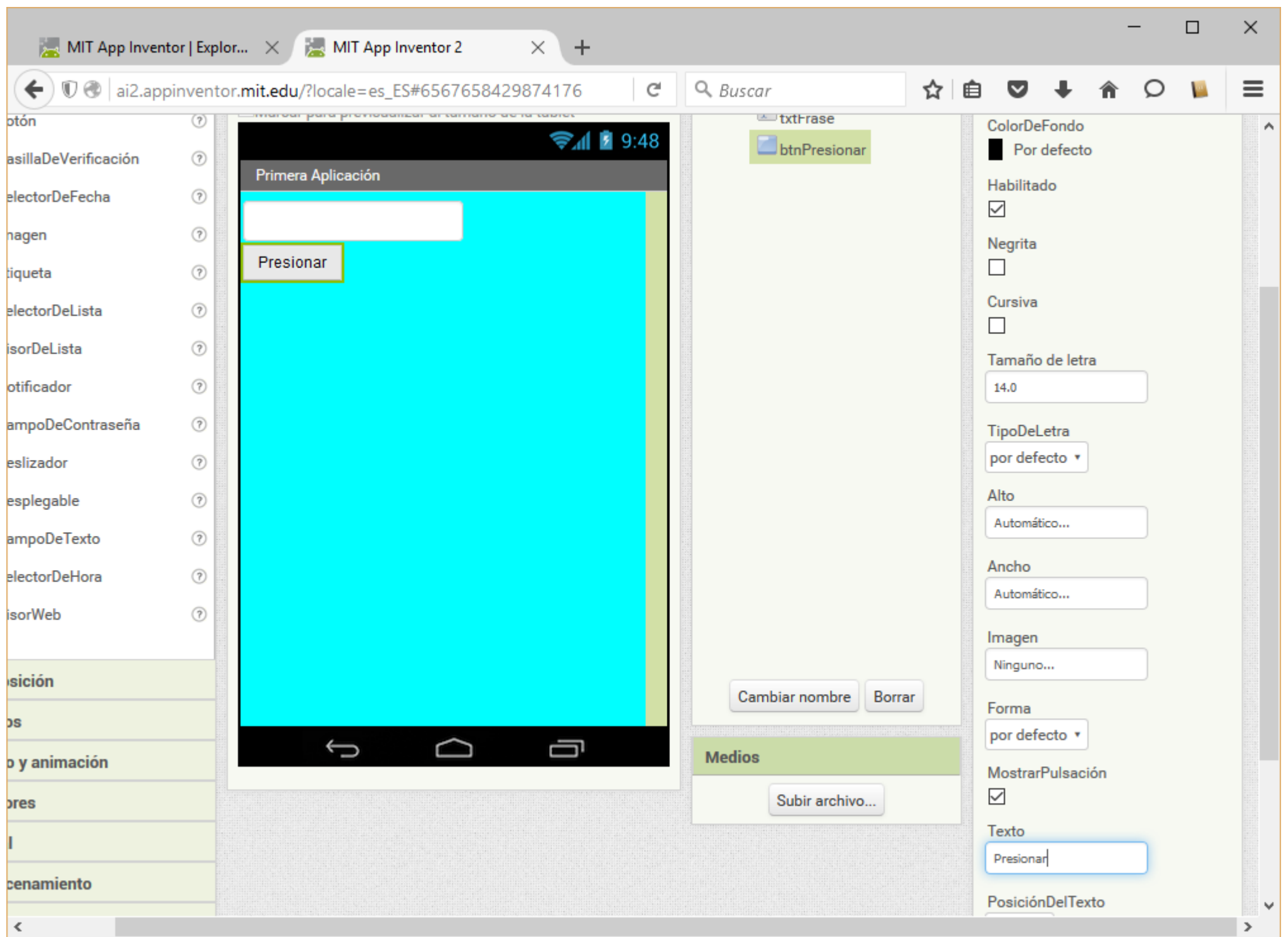


Imagen 29: Cambia las propiedades del botón

Es momento de hacer el software: El objetivo es que al presionar el botón, aparezca el texto "Hola Mundo" en el Campo de Texto. Luego se debe presionar el botón "Bloques"

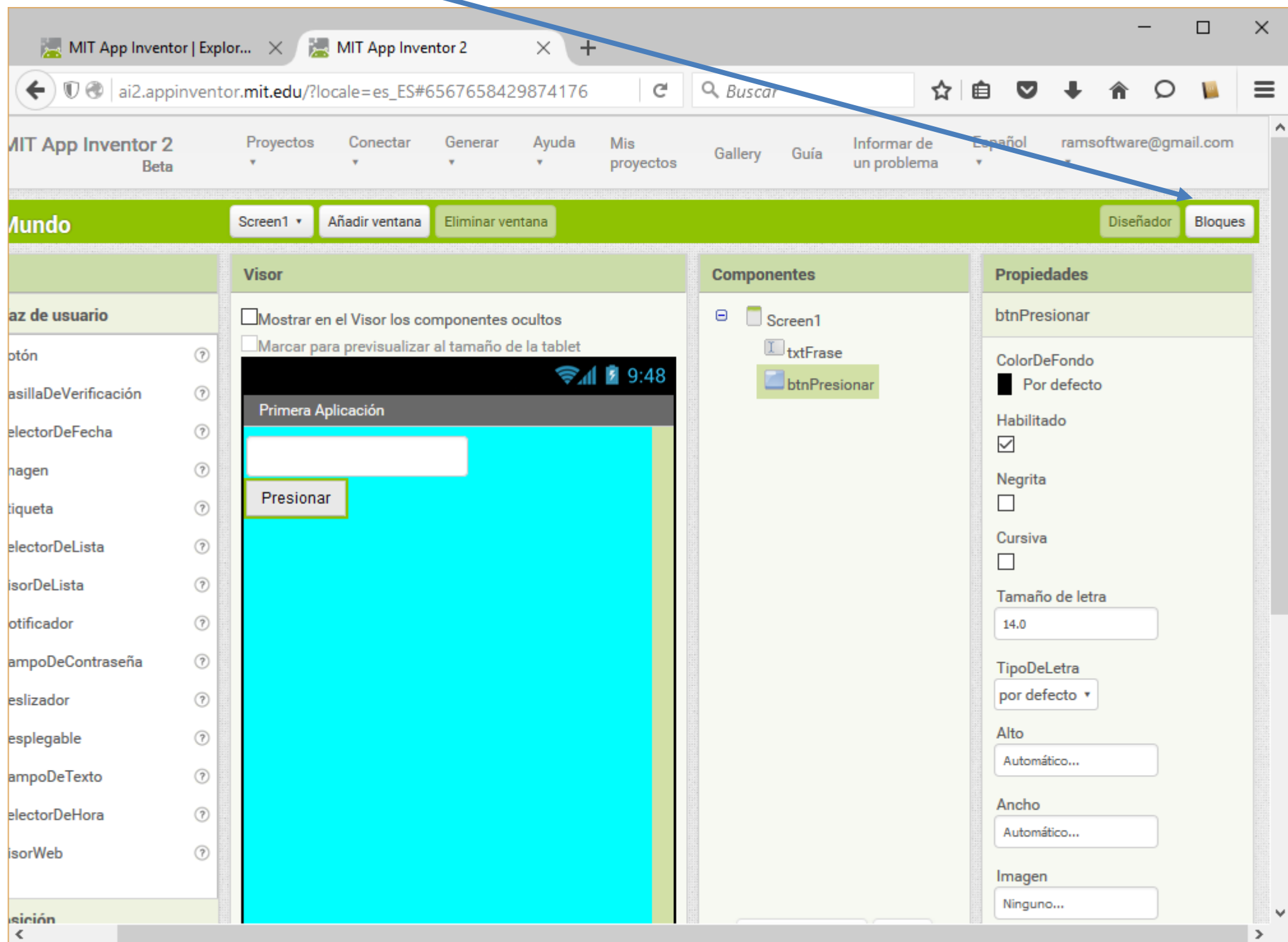


Imagen 30: Presione el botón "Bloques" para empezar a programar

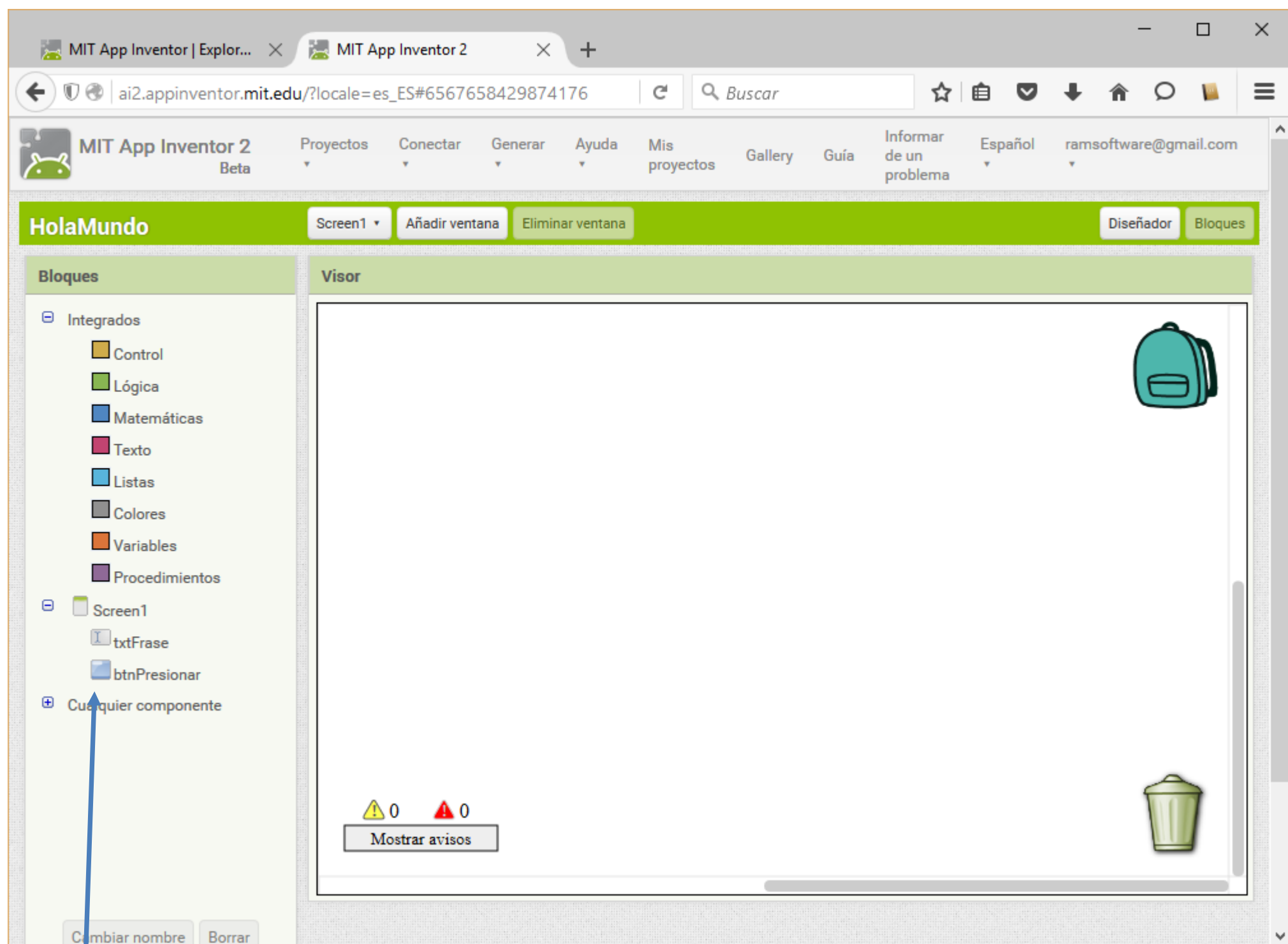


Imagen 31: Pantalla donde se programa la aplicación.

En "Bloques" aparecen los objetos visuales definidos; allí radica la importancia de nombrarlos bien con estándares:

El desarrollo es con piezas de rompecabezas que encajan. Se da clic en "btnPresionar" (un botón), y aparecen los eventos:

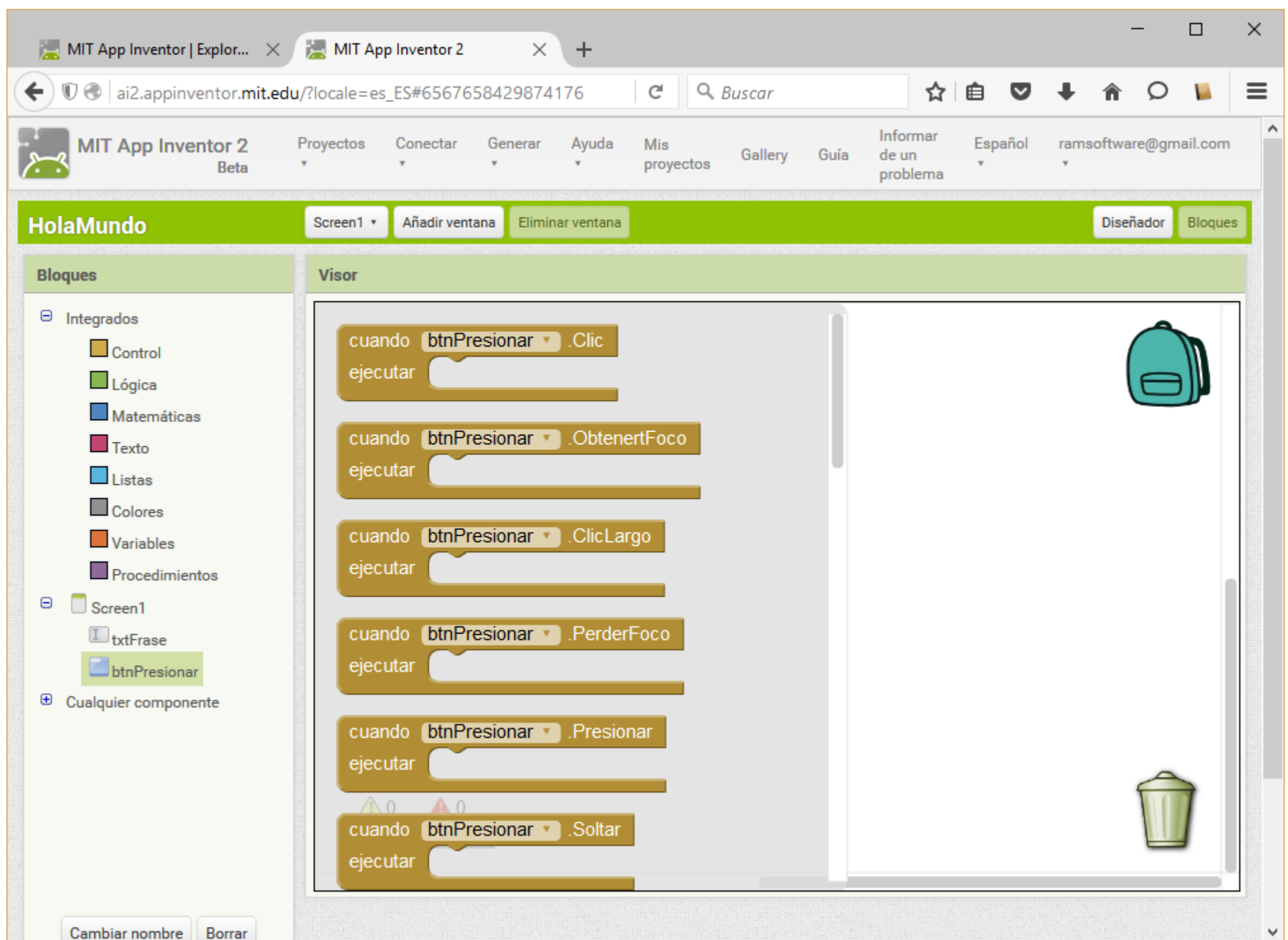


Imagen 32: La primera pieza es un evento que se dispara cuando se da clic en el botón

De clic a esa pieza del rompecabezas que representa un evento a la pantalla:

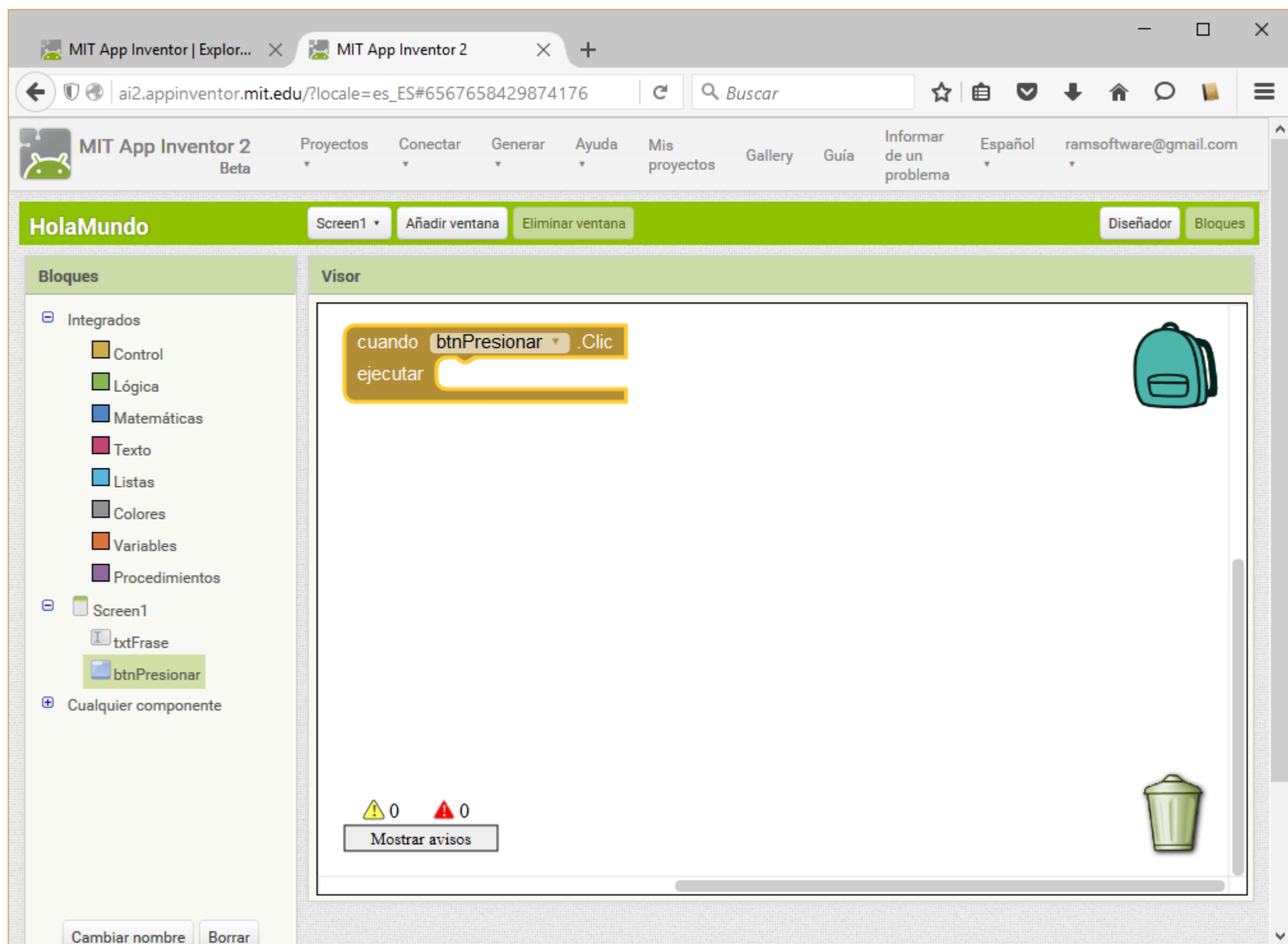


Imagen 33: Pieza que representa un evento a la pantalla

Ahora es el turno del objeto Campo de Texto, al dar clic en este, nos muestra todos los eventos, acciones y cambio de valor de propiedades que pueden hacerse:

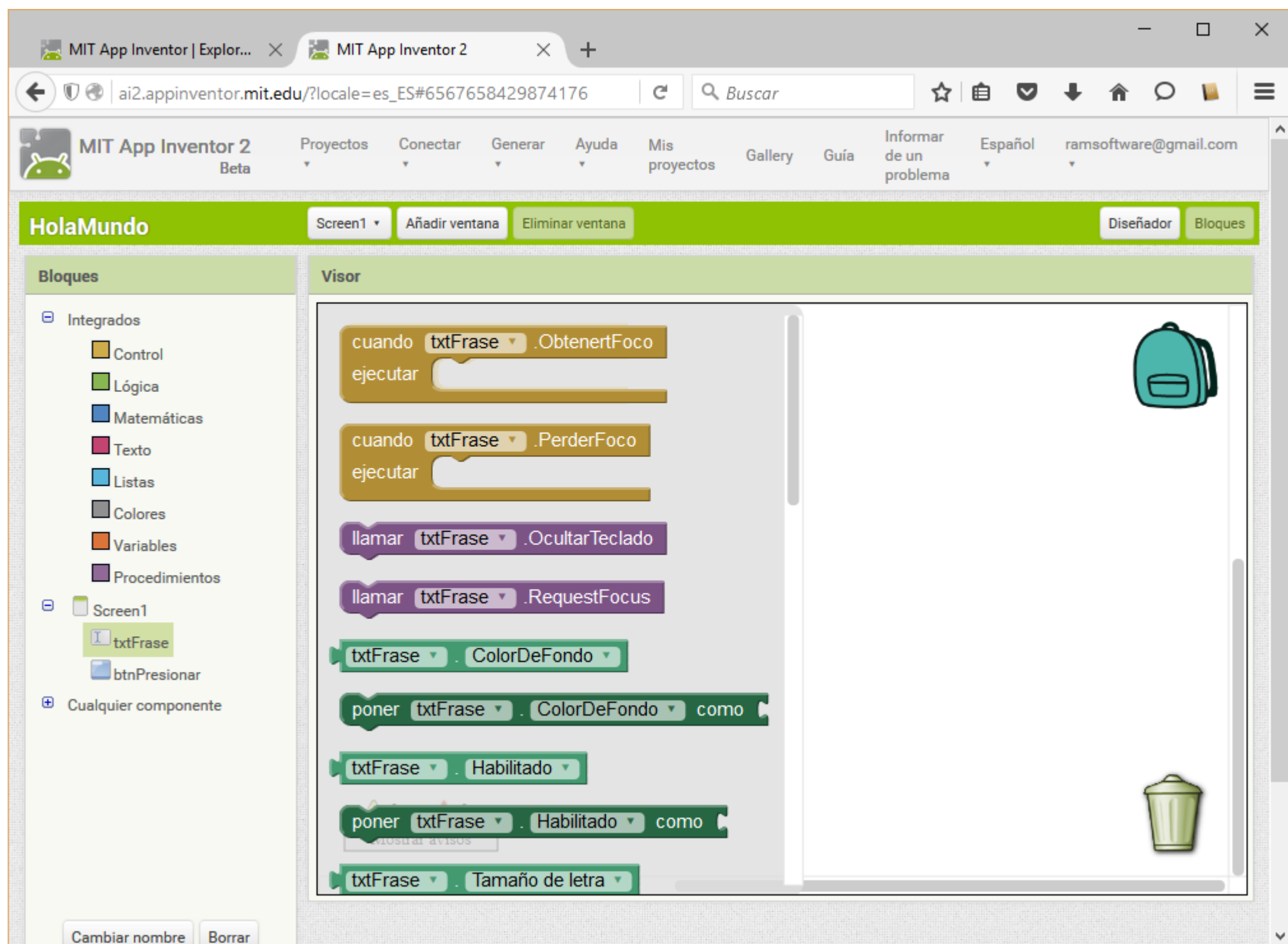


Imagen 34: Eventos del campo de texto

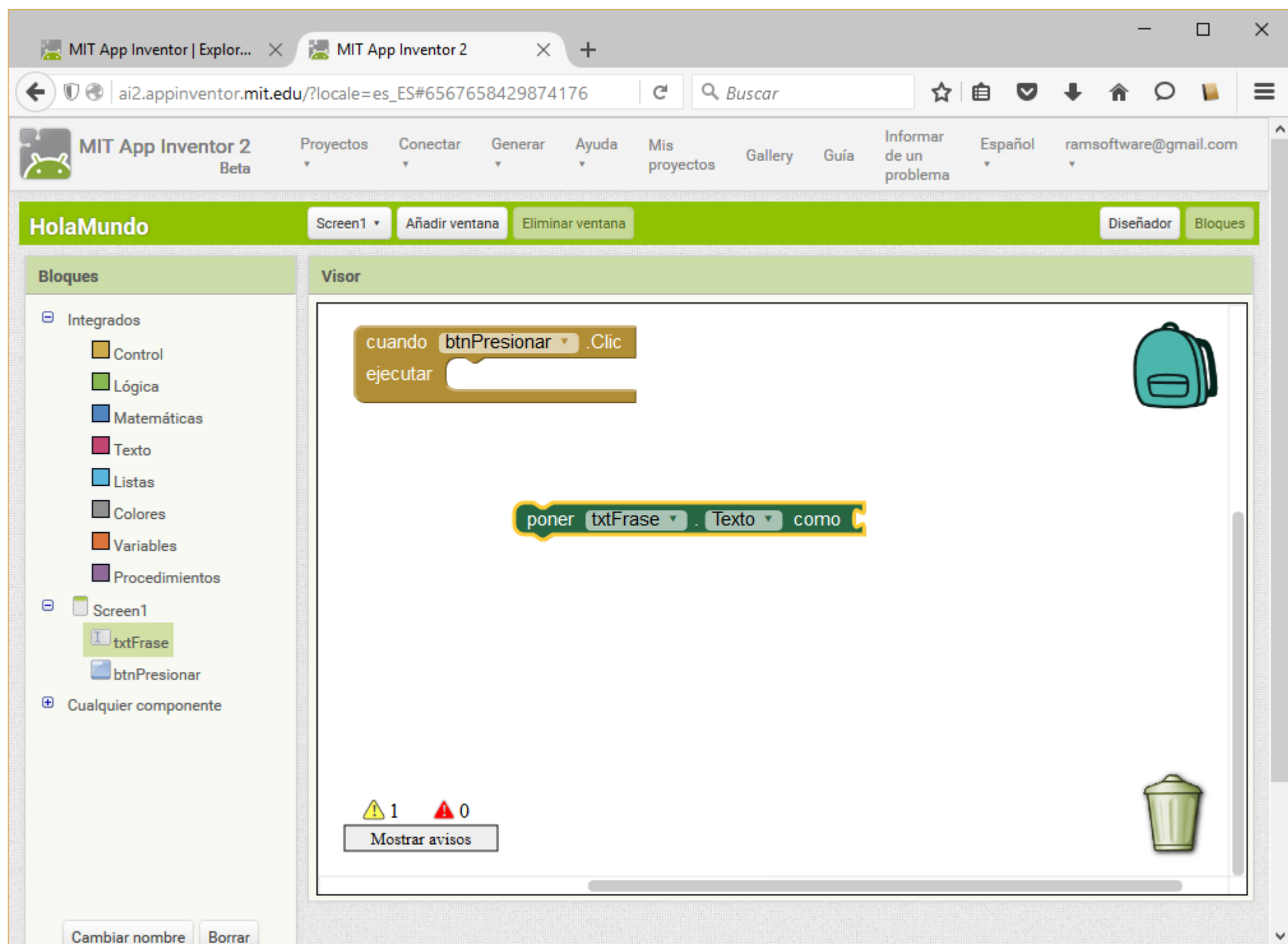


Imagen 35: Evento del campo de texto

Y observe que la pieza se acopla perfectamente al evento del botón

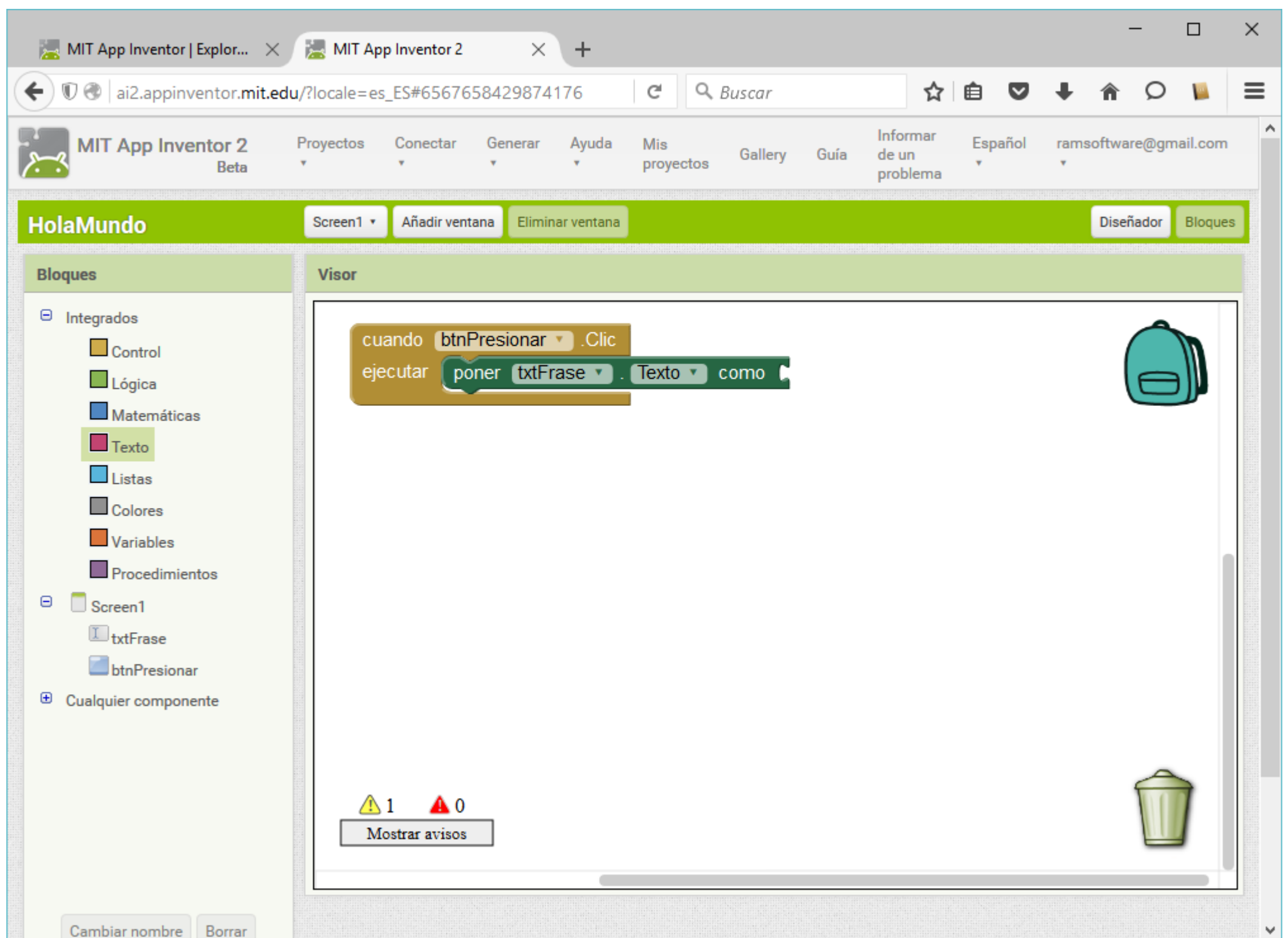


Imagen 36: Pieza se acopla al evento

Ahora es darle un valor al "campo de texto", para ello en la sección de "Integrados" seleccione "Texto" y luego clic a la primera pieza, arrástrela hasta que se acople a la pieza de cambio de valor de propiedad del "Campo de Texto"

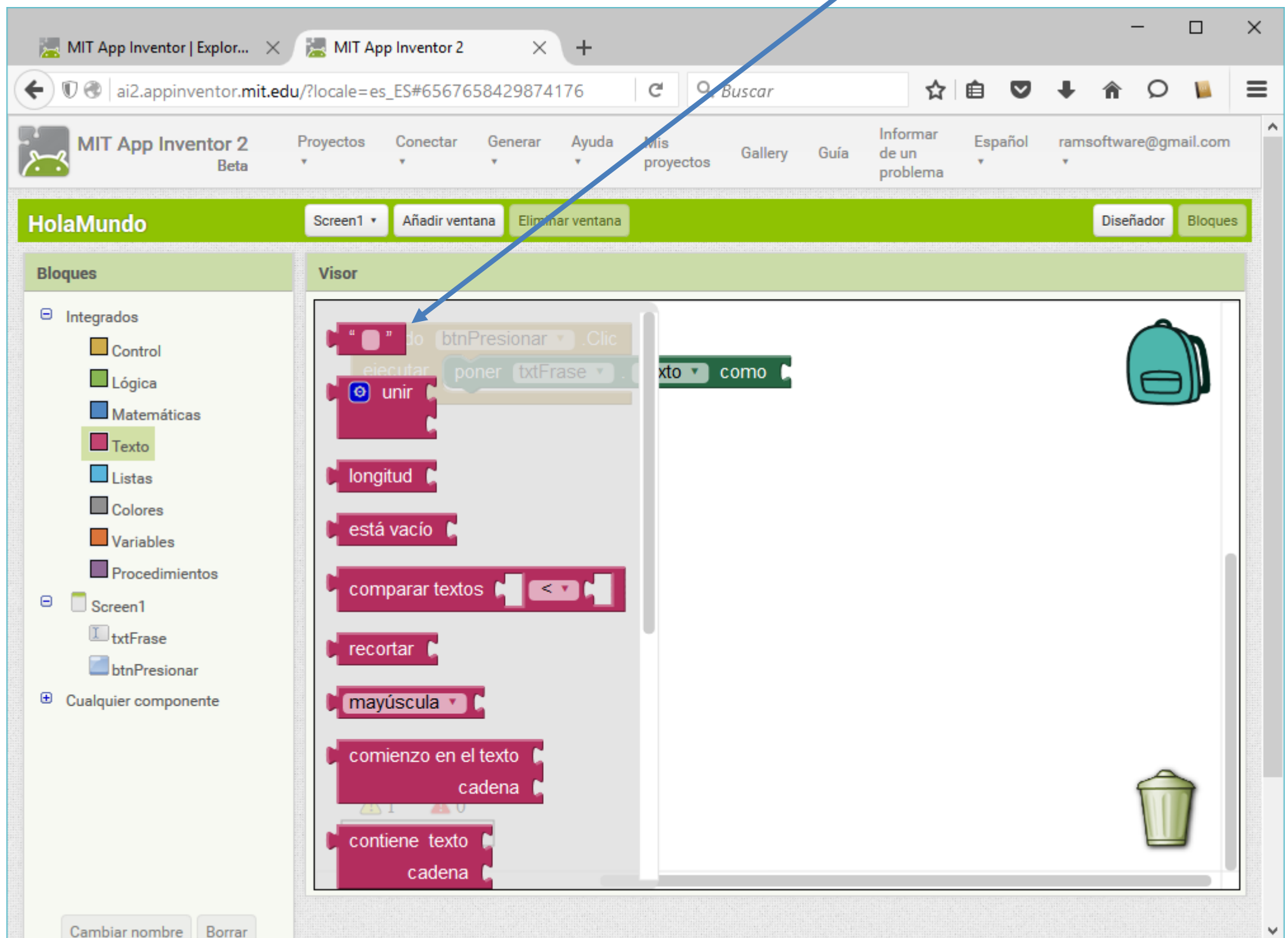


Imagen 37: Seleccione de "Integrados", en "Texto", la primera pieza

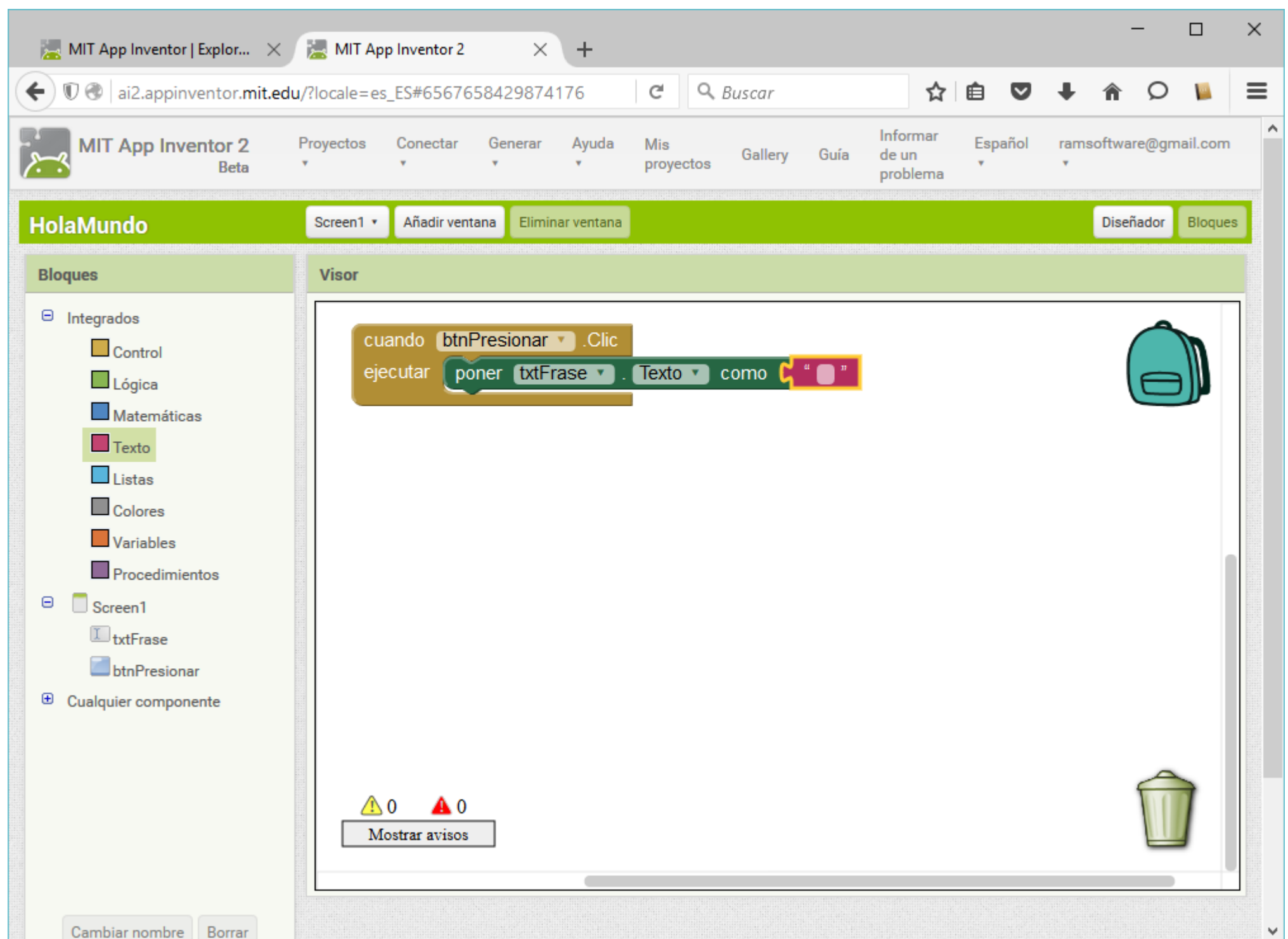


Imagen 38: Pieza acoplada

Escriba el texto "Hola Mundo" entre las comillas dobles como se ve en la imagen

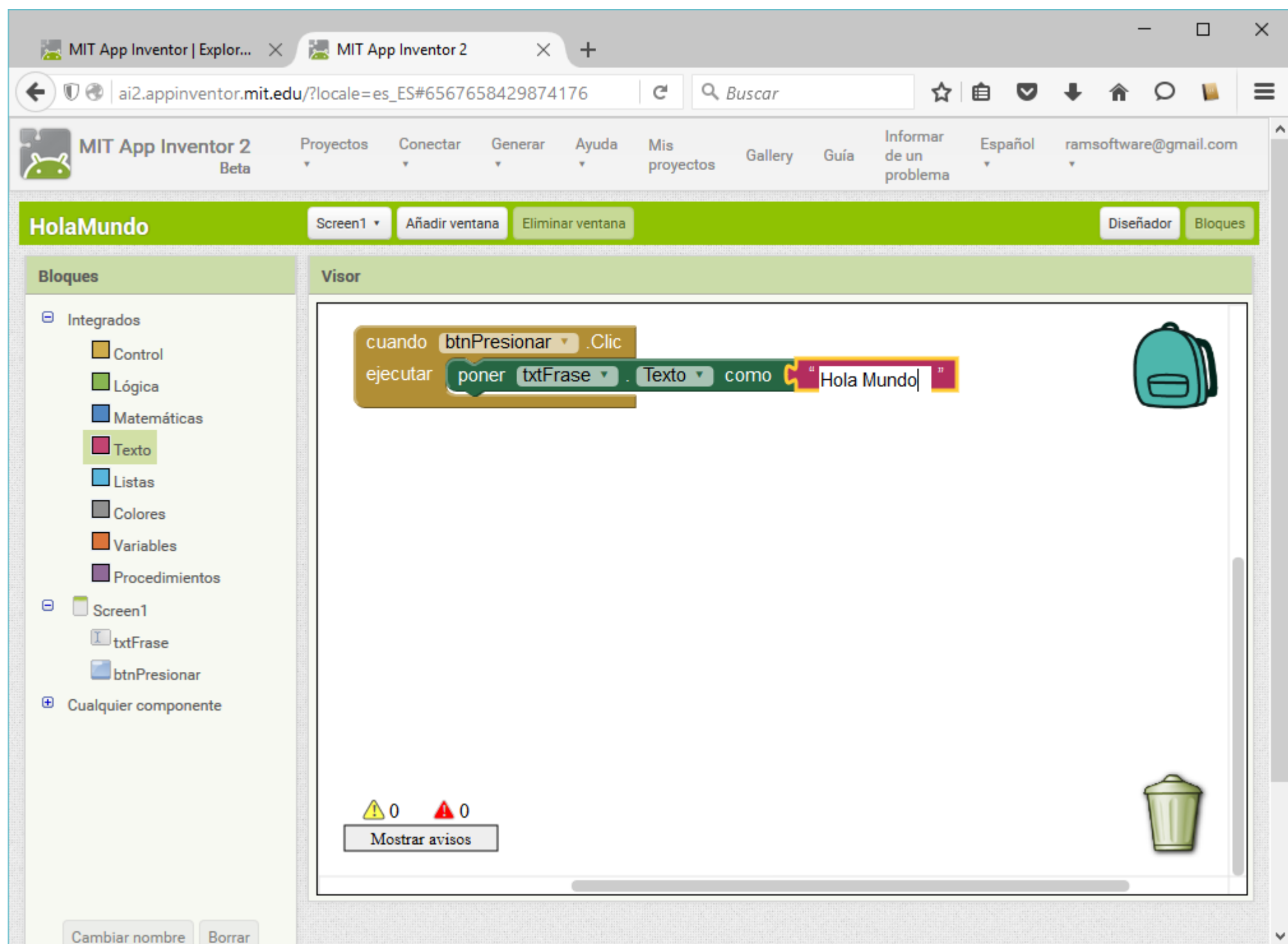


Imagen 39: Cambia el texto por el que se desea

Ya ha terminado, es momento de probar la aplicación. Si usa el emulador entonces hay que activarlo

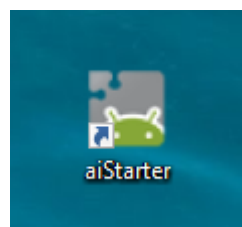


Imagen 40: Acceso directo a aiStarter

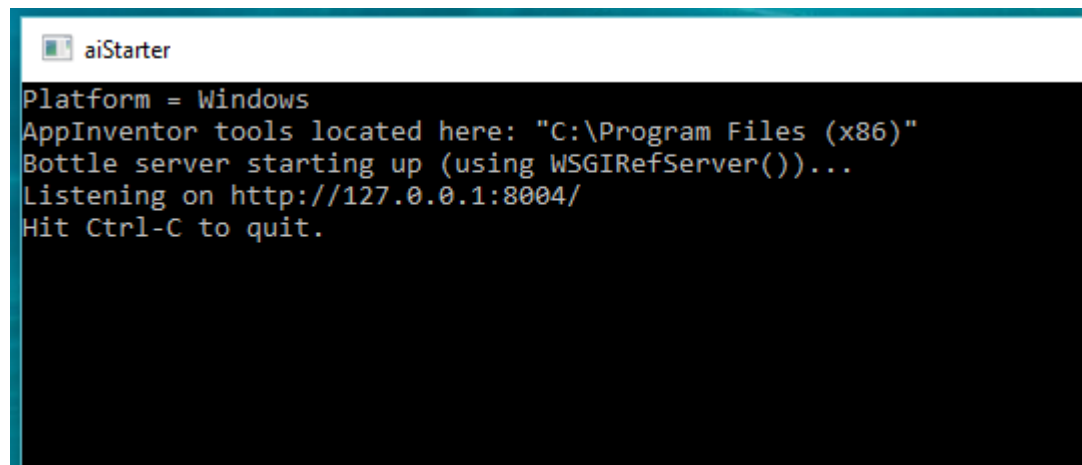


Imagen 41: aiStarter activado

Hay que dar clic en el menú de "Connect" y allí a Emulador

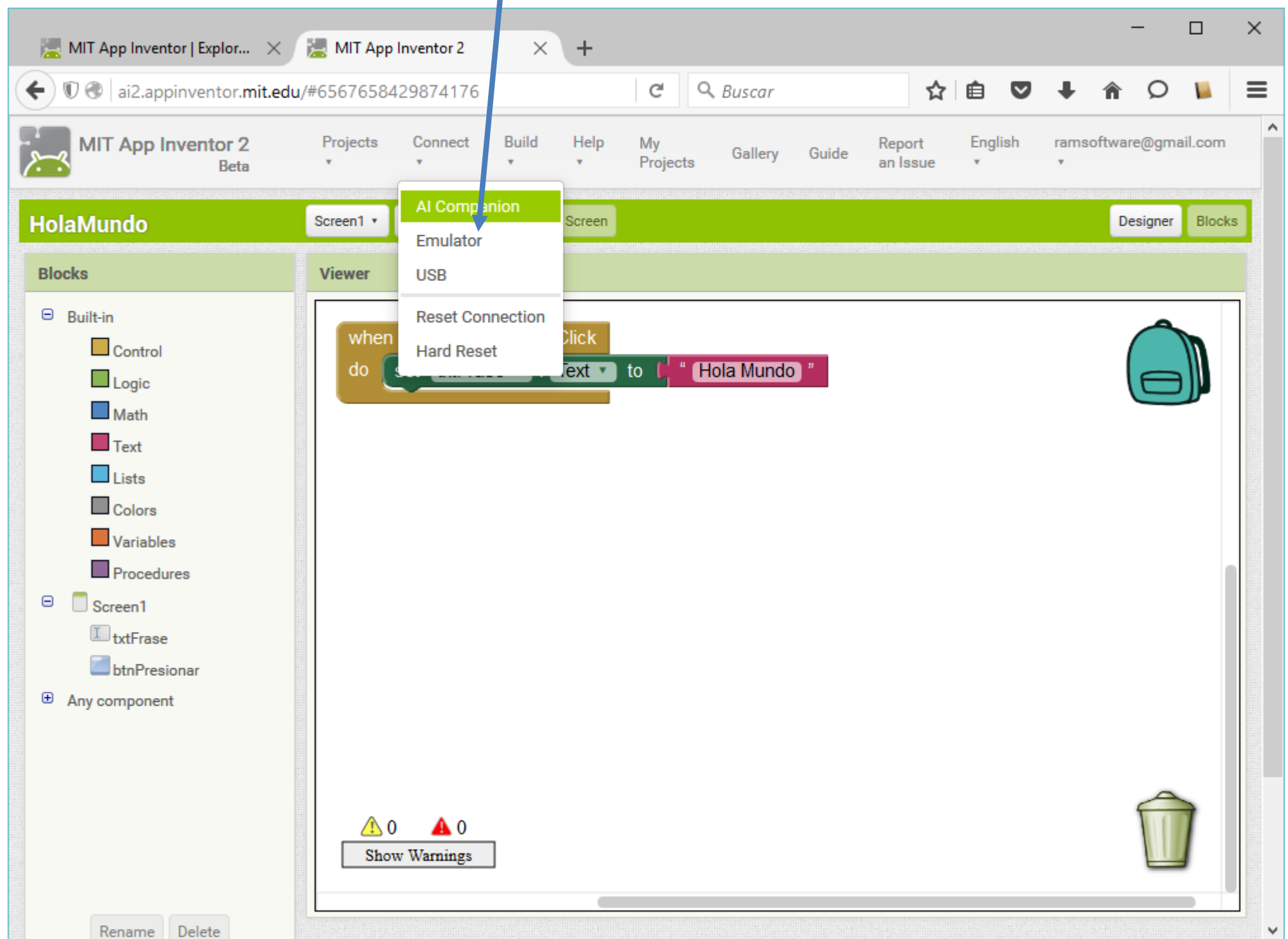


Imagen 42: Conectándose al emulador

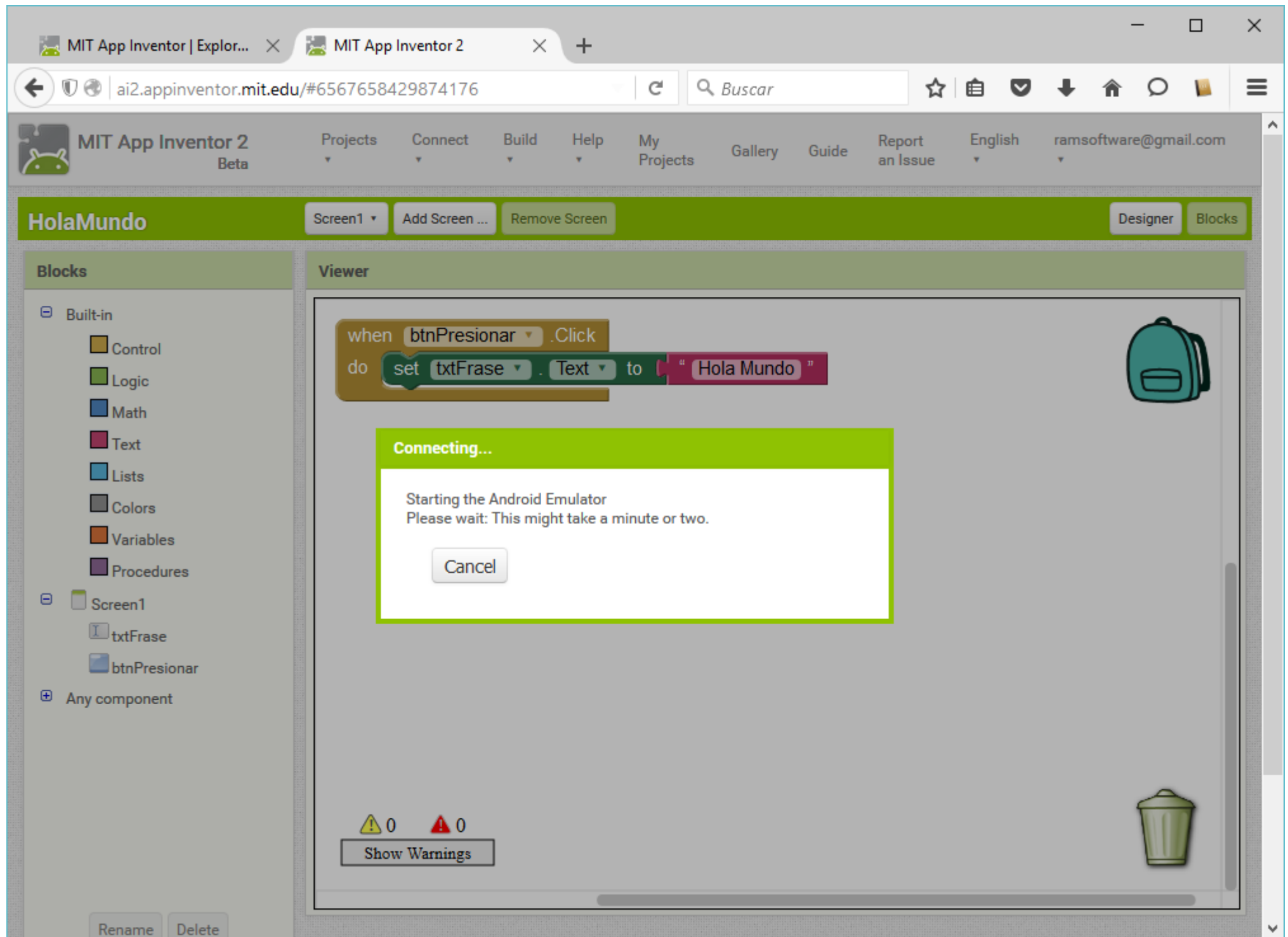


Imagen 43: Toma un minuto o dos conectarse al emulador.

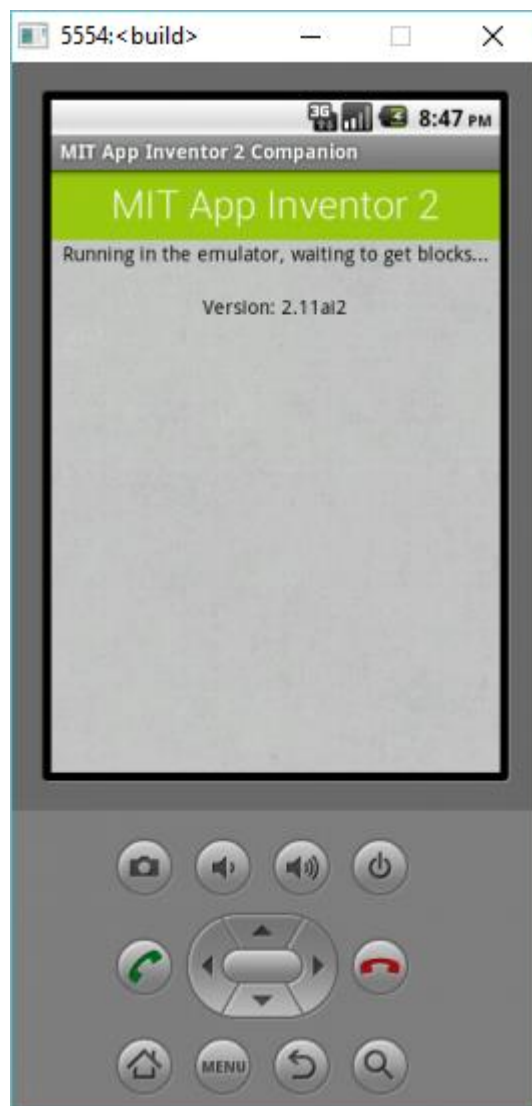


Imagen 44: El emulador ha iniciado

Una vez que el emulador ha iniciado, vuelva a la pantalla de edición de bloques. Es probable que aparezca este aviso

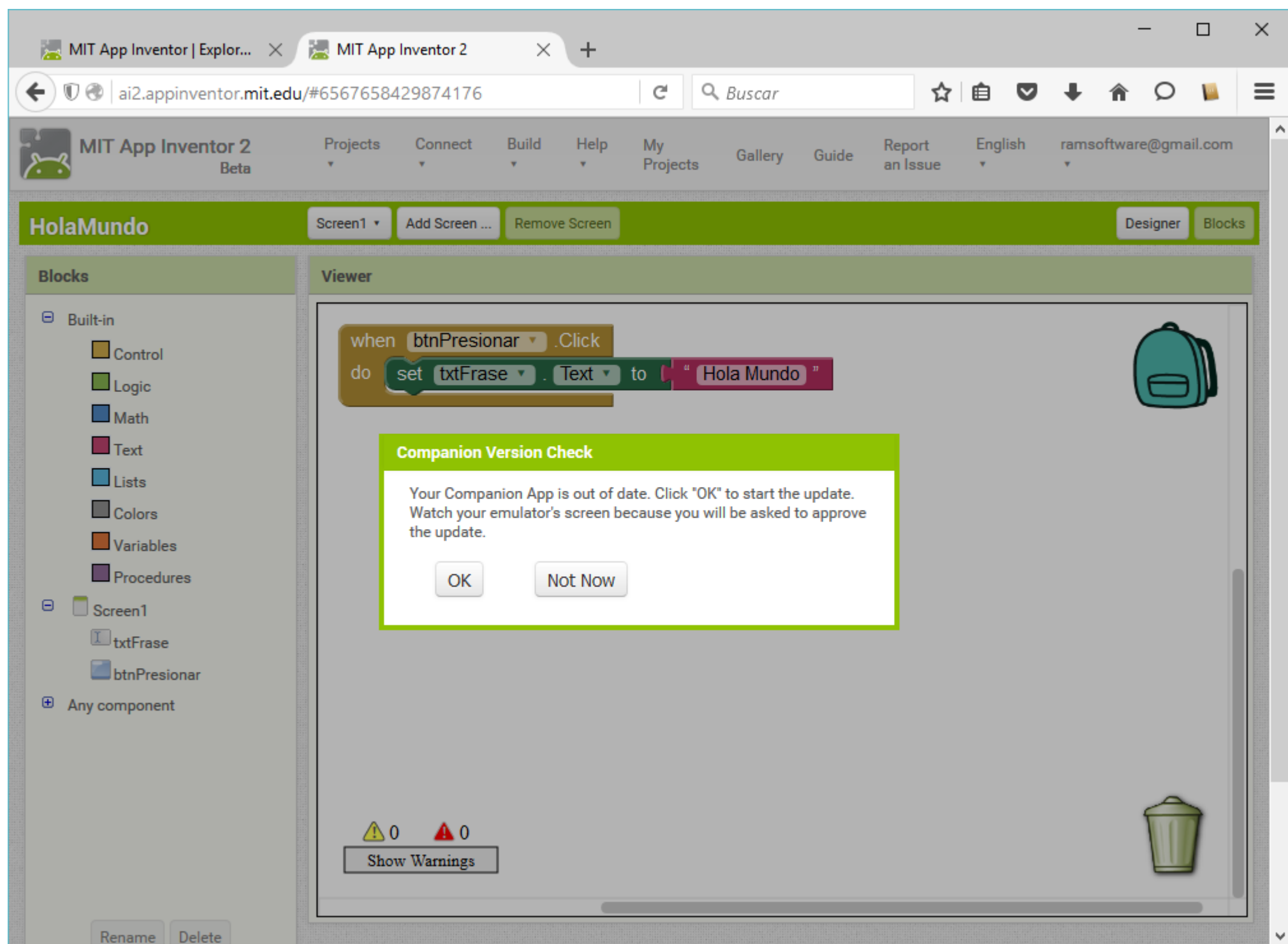


Imagen 45: La aplicación Companion dentro del emulador está obsoleta, hay que actualizarla, de clic en "OK"

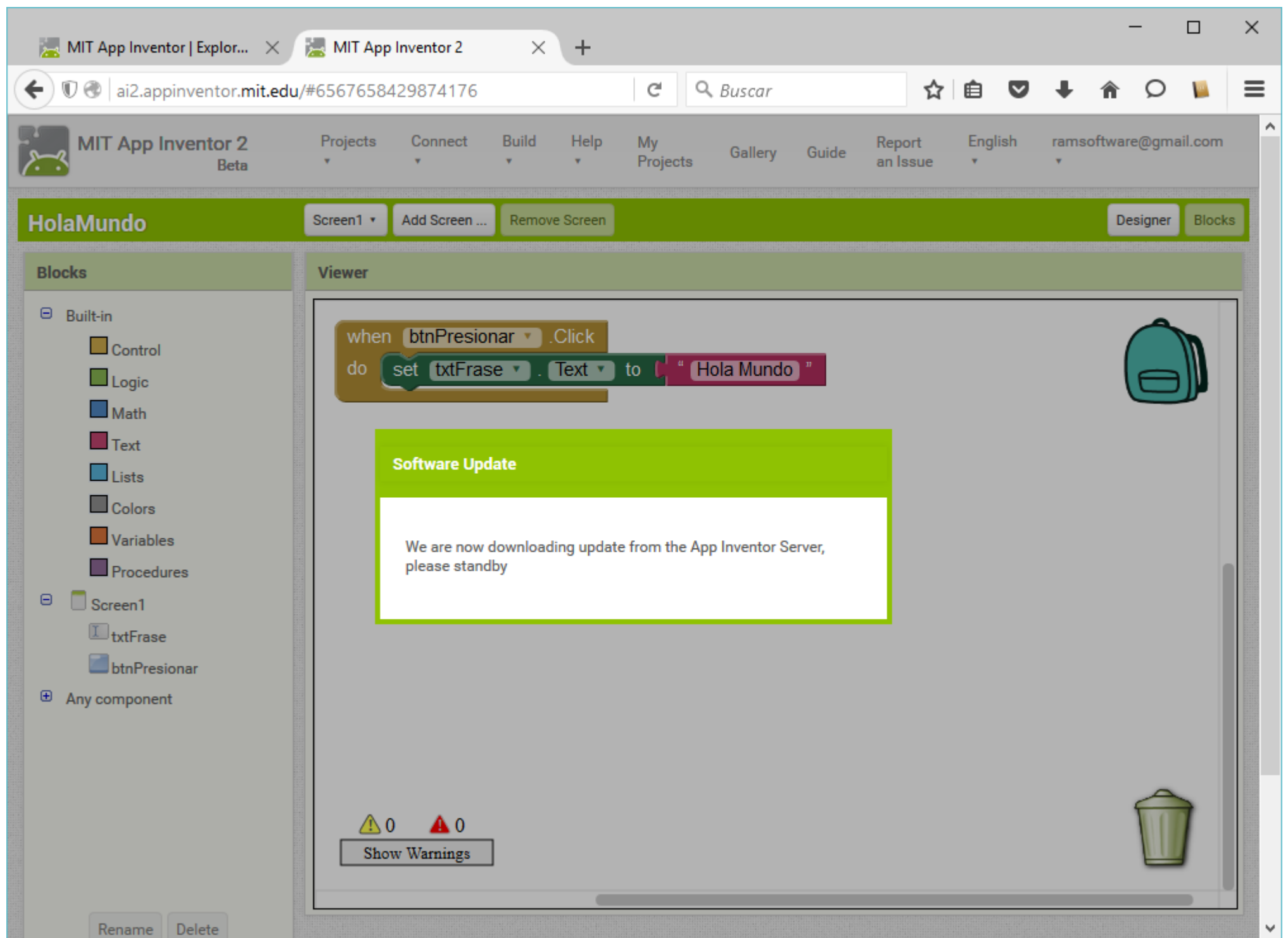


Imagen 46: Actualizando la aplicación Companion dentro del emulador

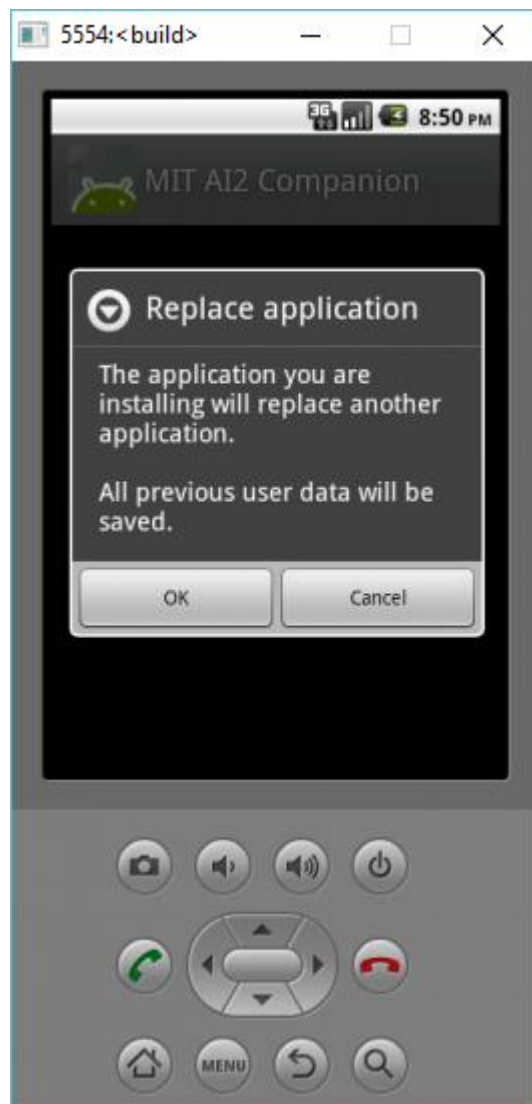


Imagen 47: Debe dar clic en "OK" para instalar la actualización

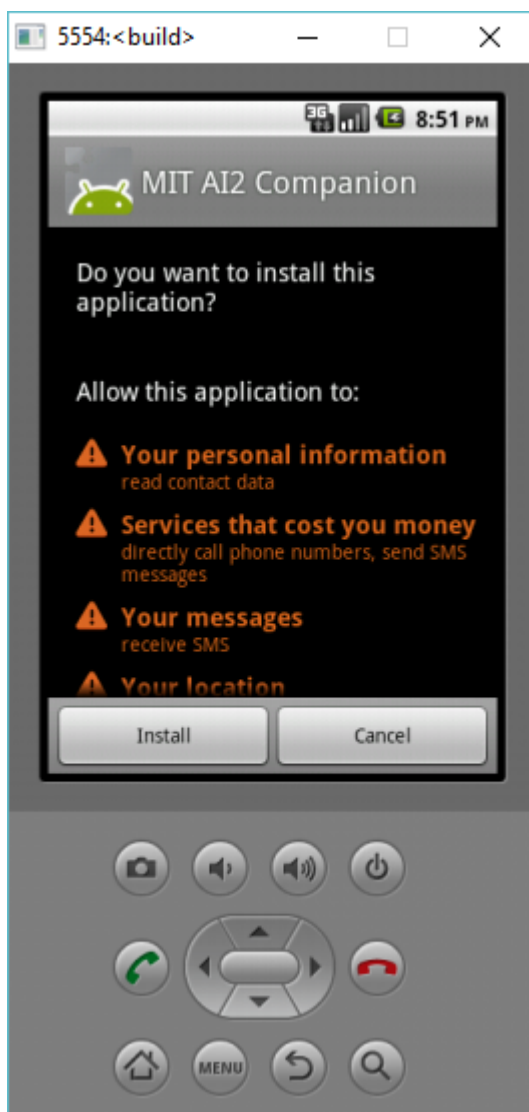


Imagen 48: Dar "Install" en la actualización

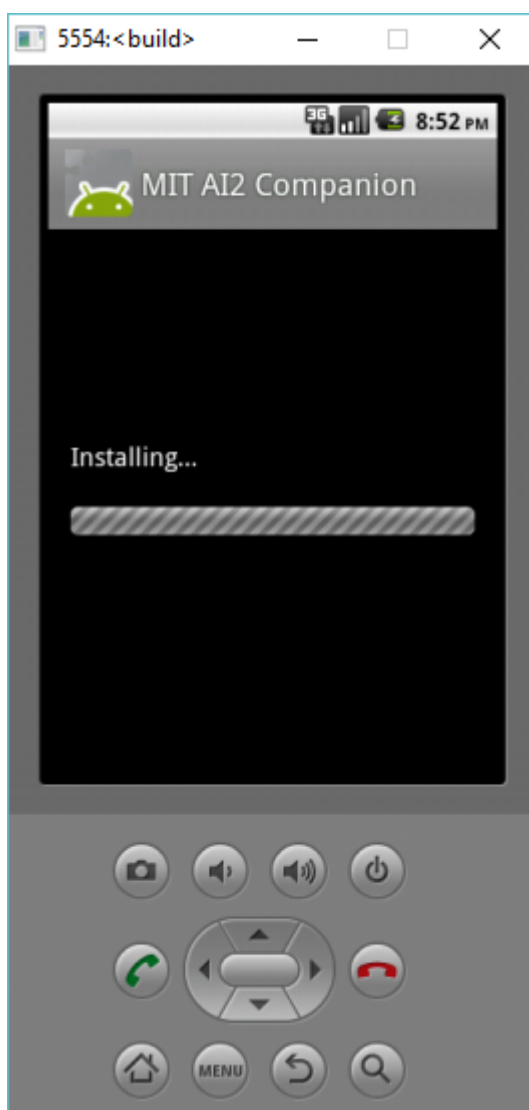


Imagen 49: Instalando la actualización en el emulador

Cuando termine de instalar, debe dar clic en "Done", **no de clic en "Open"**

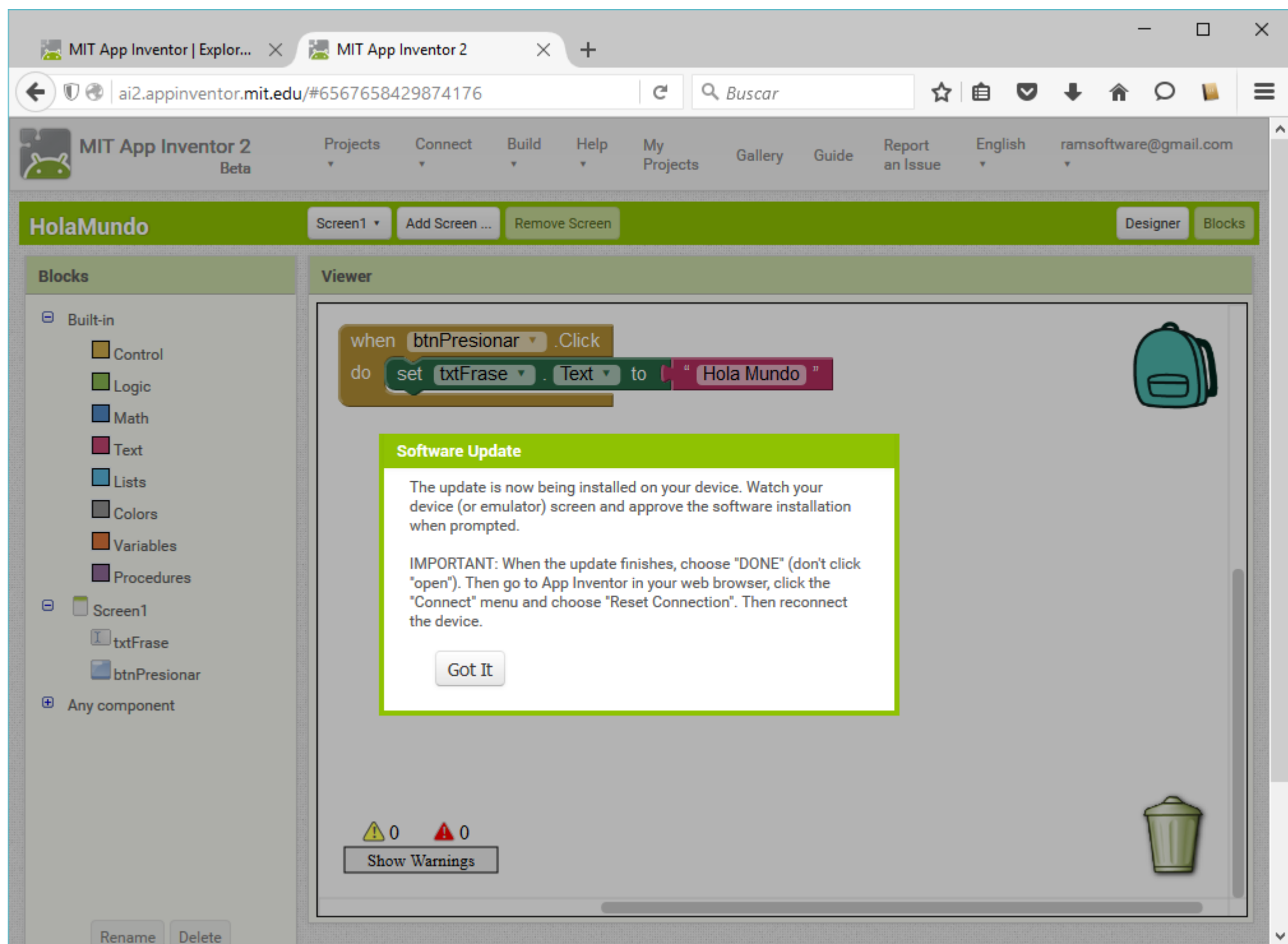


Imagen 50: Hay que dar clic en "Done", no "Open"

Luego hay que dar clic en "Reset Connection"

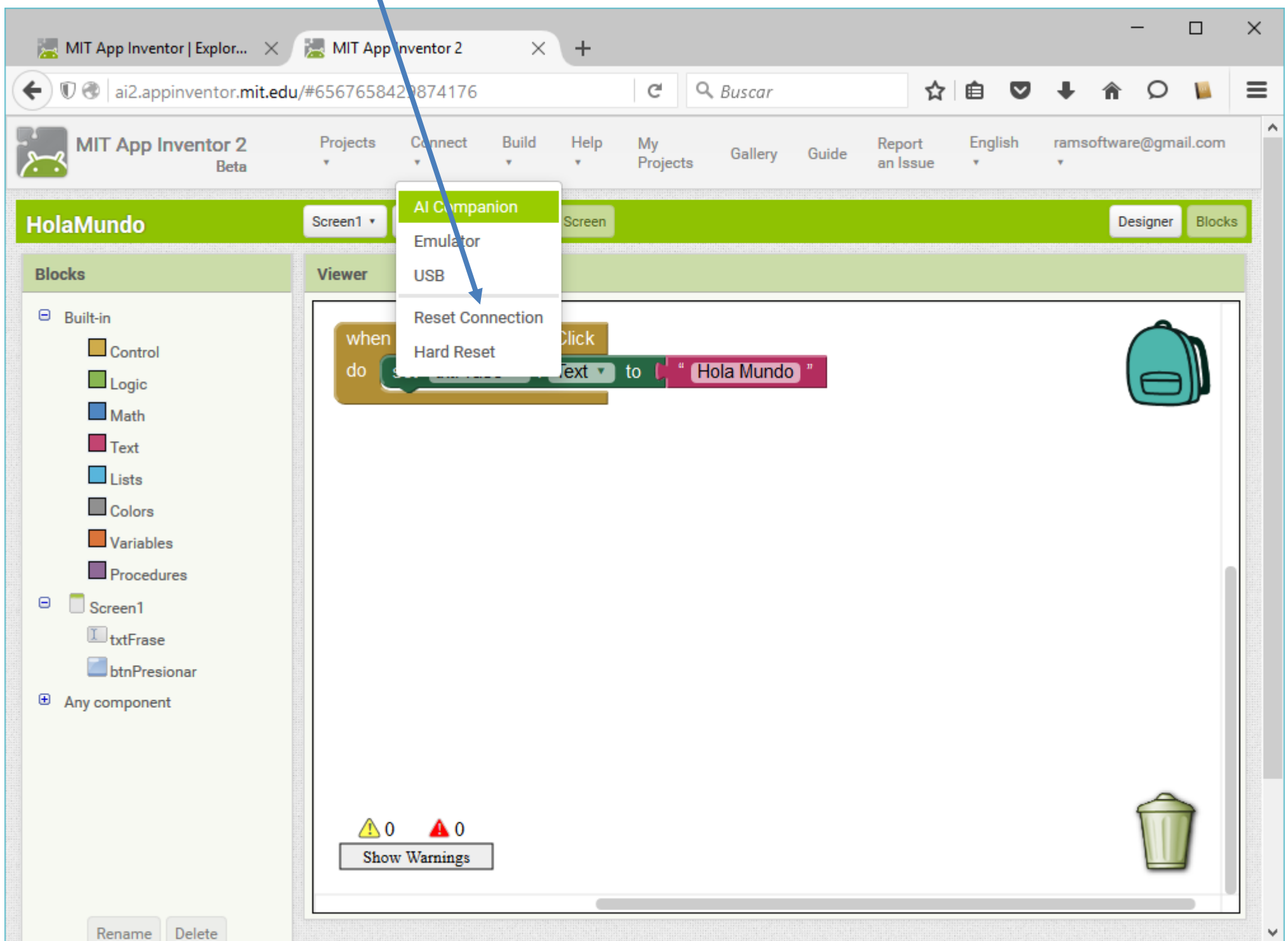


Imagen 51: Dar clic en "Reset Connection"

Al dar "Reset Connection" se cierra el emulador

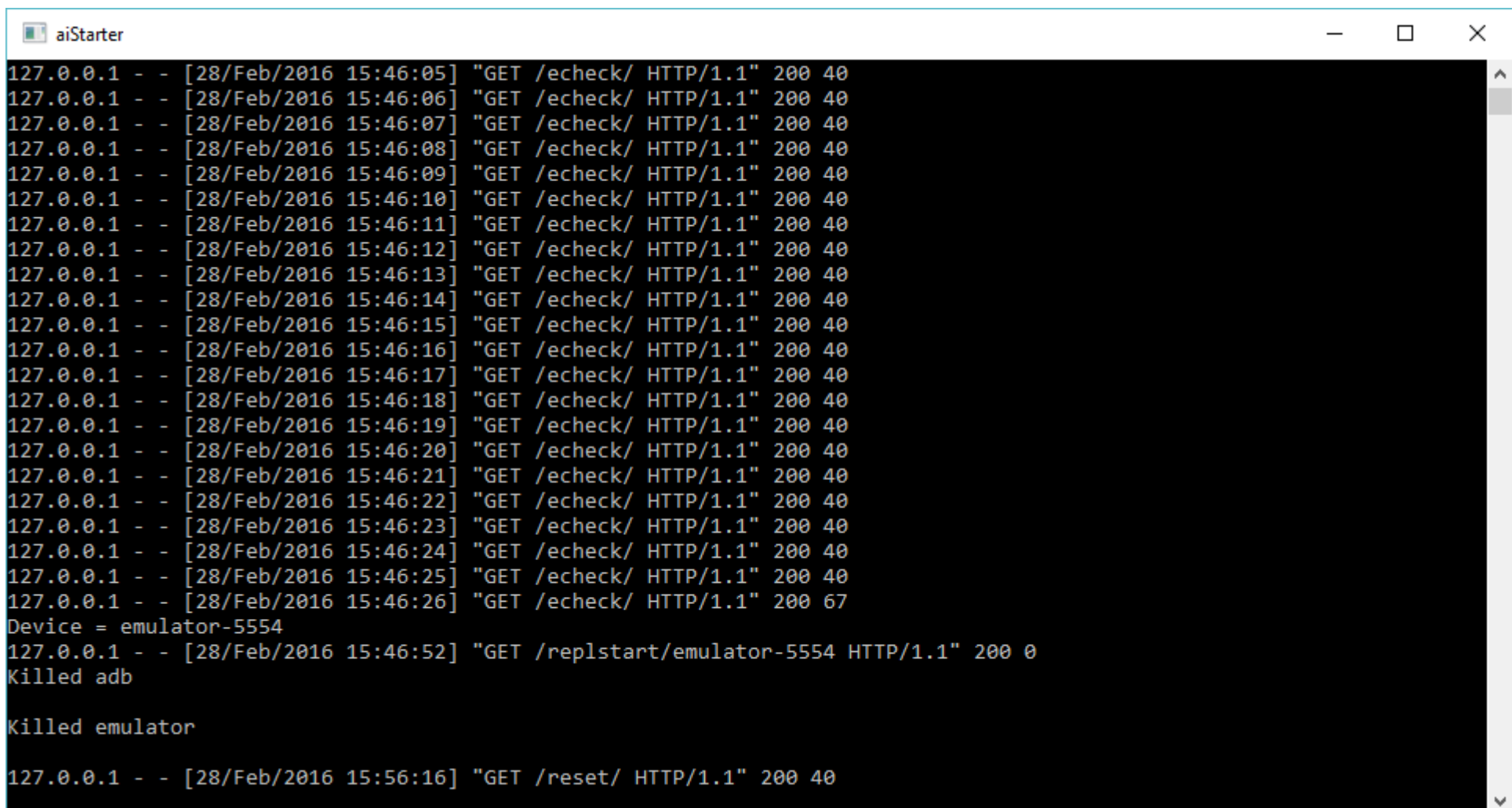


Imagen 52: Emulador se cierra

Nuevamente de clic en "Connect" y luego "Emulador"

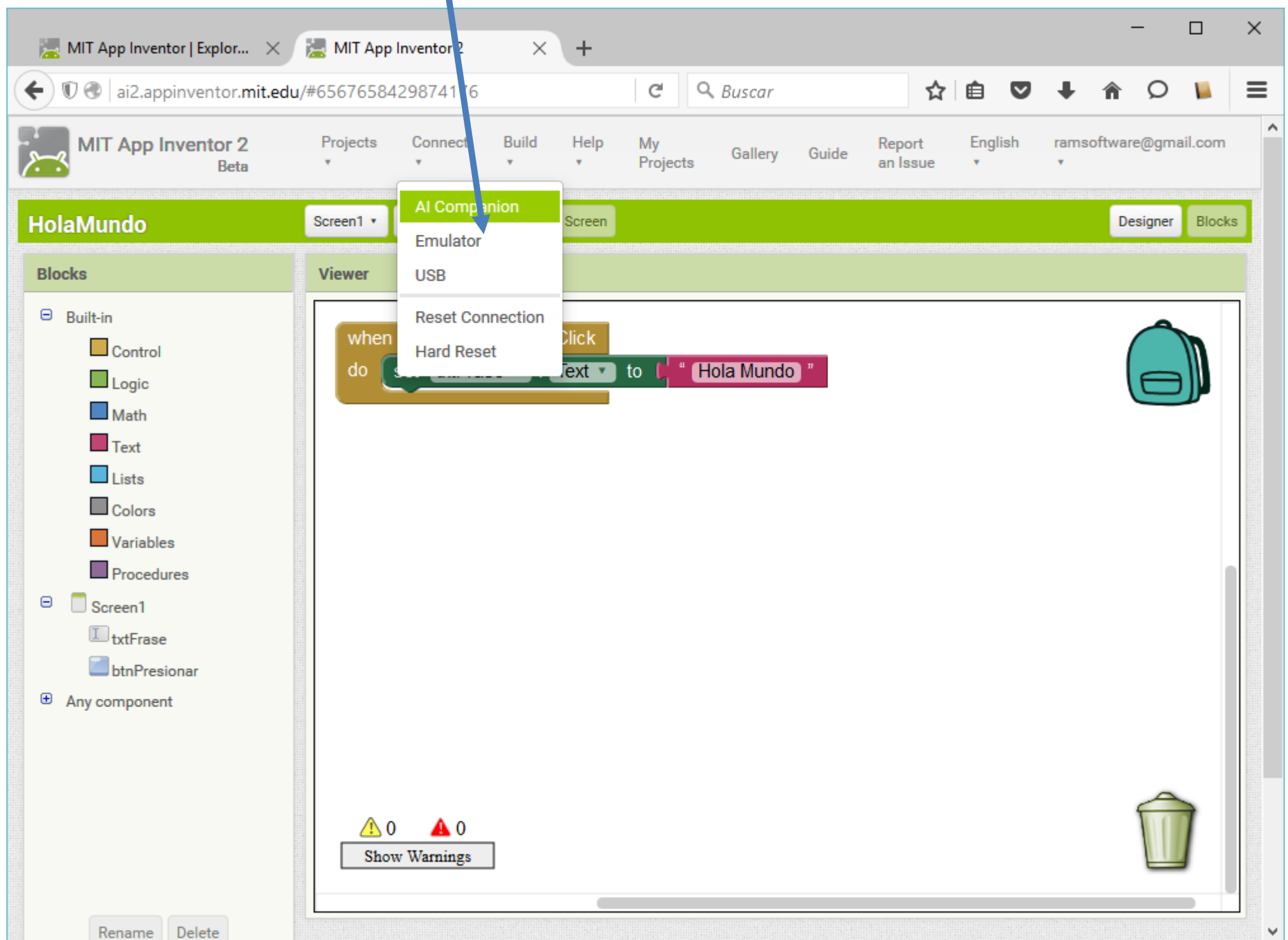


Imagen 53: De clic nuevamente en "Connect" y luego "Emulador"

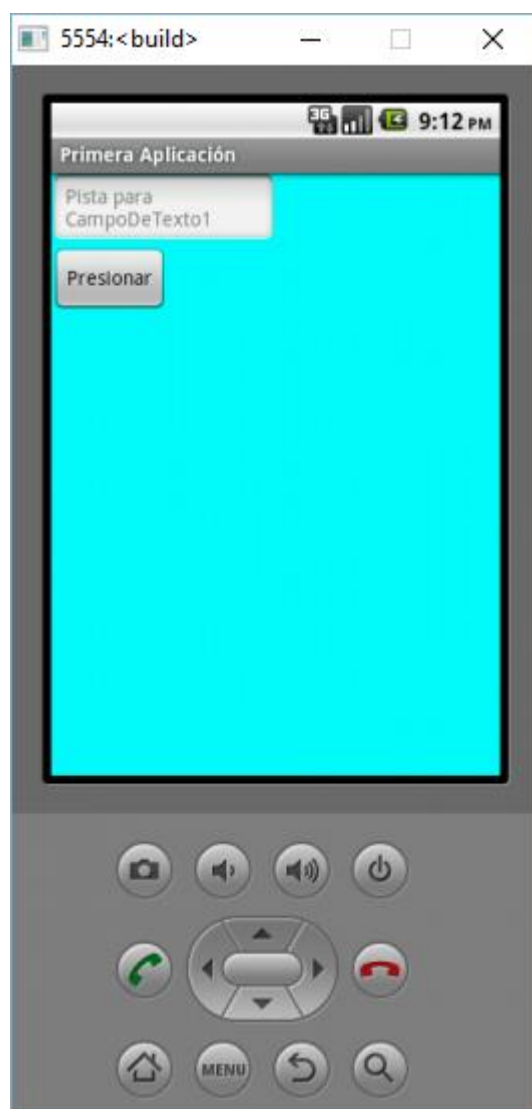


Imagen 54: Aplicación ejecutándose

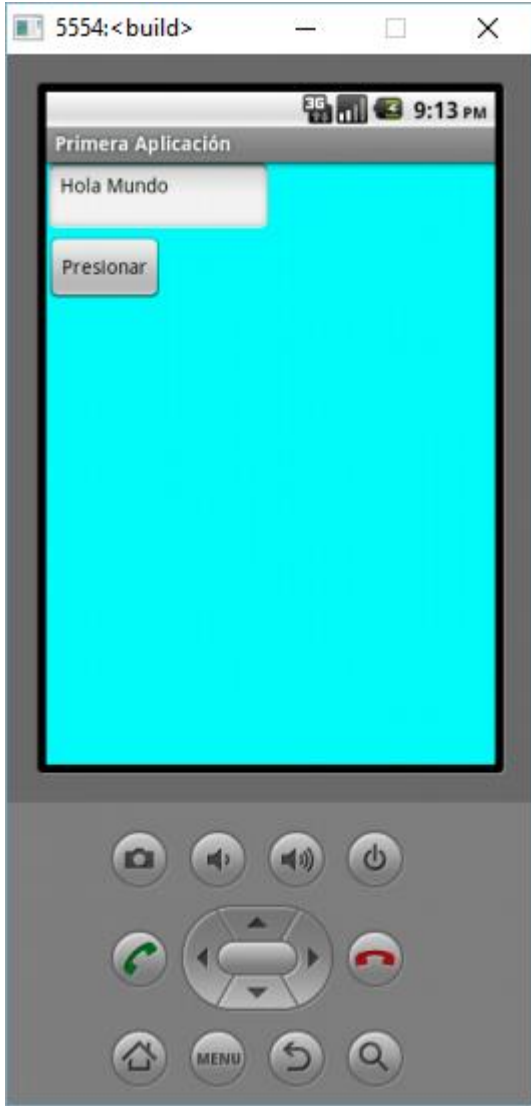


Imagen 55: Demostración que funciona la aplicación

Paso 7. Leer un par de números y luego sumarlos

Aquí aprenderá cómo es la captura de datos y una operación matemática.

Esta es la pantalla, tiene los siguientes objetos gráficos: etiquetas (Label), cajas de texto (TextBox) y un botón (Button). Ya los objetos fueron nombrados con el estándar.

Objeto que usará	Prefijo y forma de nombrar	Ejemplos
Botón (Button)	btn + Letra mayúscula + resto de nombre	btnAceptar, btnPresionar, btnProcesar
Caja de texto (TextBox)	txt + Letra mayúscula + resto de nombre	txtAviso, txtNombre, txtApellido
Etiqueta (Label)	lbl + Letra mayúscula + resto de nombre	lblNumeroA, lblResultado

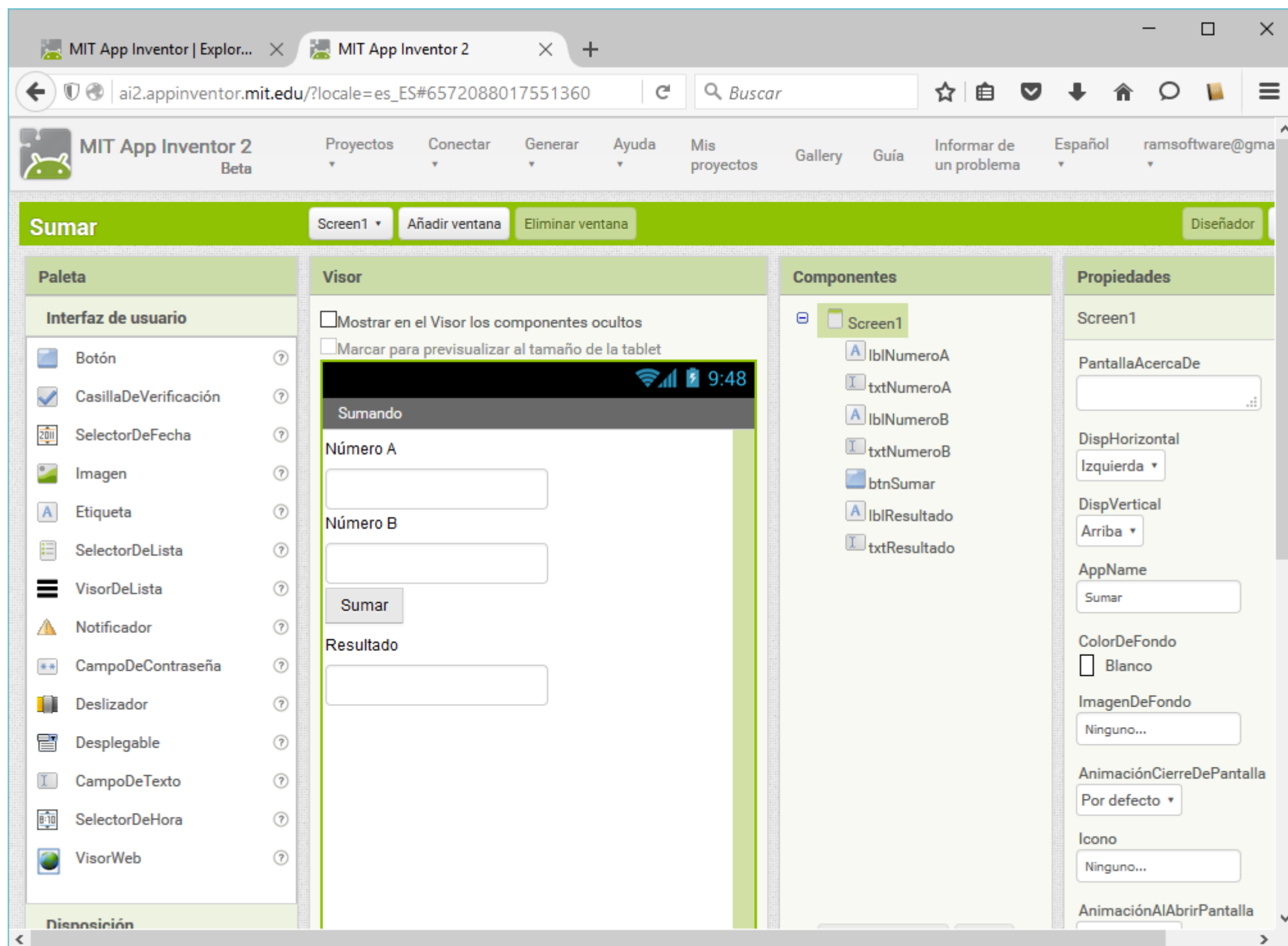


Imagen 56: Diseño de página para sumar dos números y presentar el resultado

Este es el diagrama de bloques

The image shows the MIT App Inventor 2 web interface. The browser address bar displays `ai2.appinventor.mit.edu/?locale=es_ES#6572088017551360`. The page title is "MIT App Inventor 2 Beta". The main navigation bar includes "Proyectos", "Conectar", "Generar", "Ayuda", "Mis proyectos", "Gallery", "Guía", "Informar de un problema", "Español", and "ramsoftware@gmail.com".

The interface is divided into two main sections: "Bloques" (Blocks) on the left and "Visor" (Viewer) on the right. The "Bloques" section shows a list of components under "Integrados" (Control, Lógica, Matemáticas, Texto, Listas, Colores, Variables, Procedimientos) and "Screen1" (lblNumeroA, txtNumeroA, lblNumeroB, txtNumeroB, btnSumar, lblResultado, txtResultado). The "Visor" section displays the block diagram for the "Sumar" application. The diagram consists of the following blocks:

- Three "inicializar global" blocks: "inicializar global valA como 0", "inicializar global valB como 0", and "inicializar global Resultado como 0".
- A "cuando btnSumar .Clic" block containing an "ejecutar" block with the following sub-blocks:
 - "poner global valA a txtNumeroA . Texto"
 - "poner global valB a txtNumeroB . Texto"
 - "poner global Resultado a tomar global valA + tomar global valB"
 - "poner txtResultado . Texto como tomar global Resultado"

At the bottom of the "Visor" section, there are two warning icons (yellow and red triangles) with a "0" next to each, and a "Mostrar avisos" button. A trash can icon is also visible in the bottom right corner of the viewer area.

Imagen 57: Diagrama de bloques para sumar dos números

Explicación: Primero se definen las variables en este menú y usando esta opción

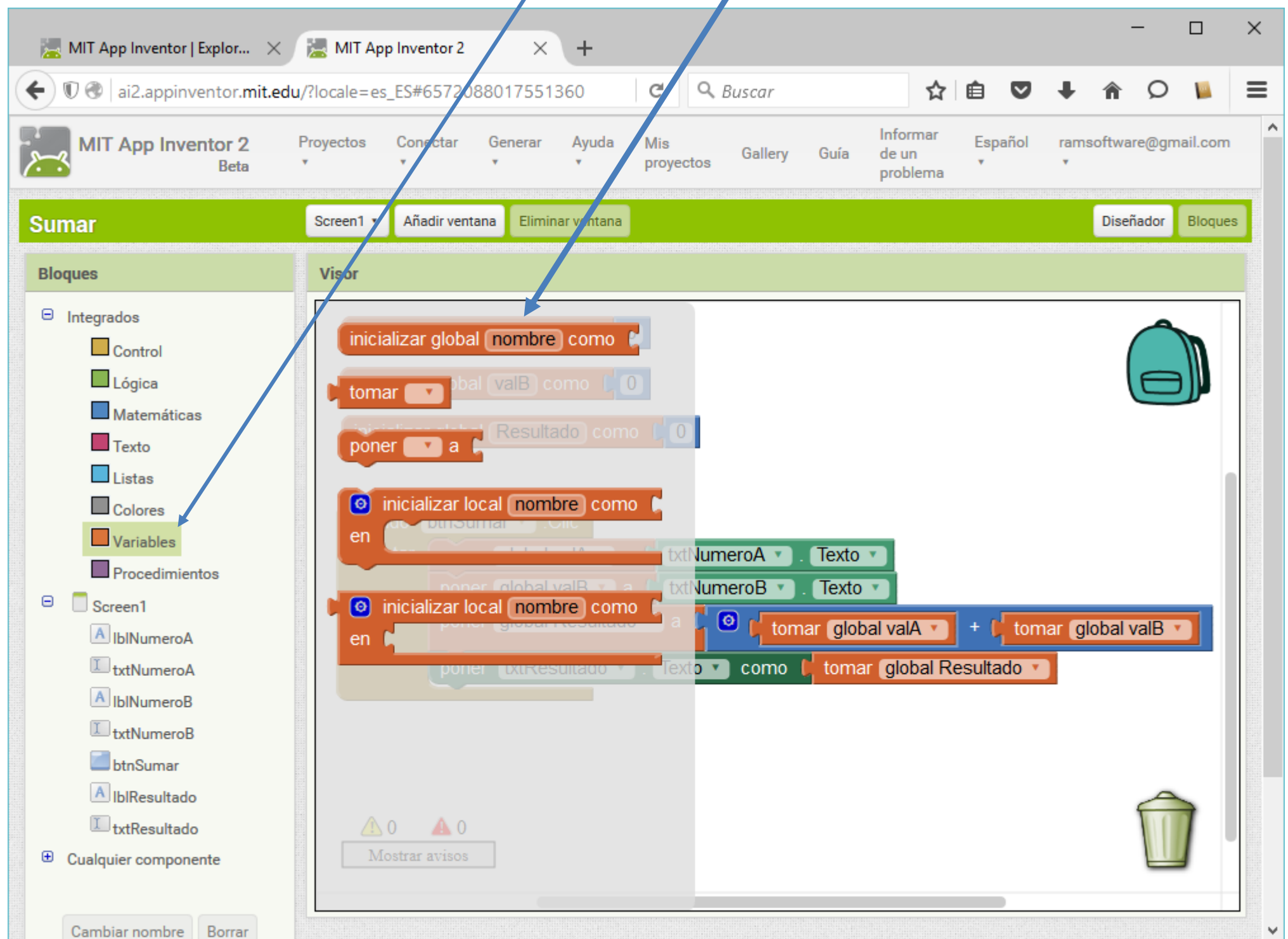


Imagen 58: Menú para definir variables

Segundo se inicializan esas variables con este menú y se dan valores con esta opción

The screenshot displays the MIT App Inventor 2 web interface. The browser address bar shows the URL `ai2.appinventor.mit.edu/?locale=es_ES#6572088017551360`. The page title is "MIT App Inventor 2 Beta". The main navigation bar includes "Proyectos", "Conectar", "Generar", "Ayuda", "Mis proyectos", "Gallery", "Guía", "Informar de un problema", "Español", and "ramsoftware@gmail.com". The current project is named "Sumar".

The interface is divided into two main sections: "Bloques" (Blocks) on the left and "Visor" (Viewer) on the right. The "Bloques" section is organized into categories: "Integrados" (Control, Lógica, Matemáticas, Texto, Listas, Colores, Variables, Procedimientos) and "Screen1" (labeled with 'A' icons). Under "Screen1", the following components are listed: `lblNumeroA`, `txtNumeroA`, `lblNumeroB`, `txtNumeroB`, `btnSumar`, `lblResultado`, and `txtResultado`. The "Matemáticas" category is highlighted in green.

The "Visor" section shows a visual programming workspace with several blocks. The top blocks are for initializing global variables: `inicializar global valA como 0`, `inicializar global valB como 0`, and `inicializar global Resultado como 0`. Below these are arithmetic blocks: `+`, `-`, `*`, and `/`. A block for generating a random integer is shown: `entero aleatorio entre 1 y 100`. Other blocks include `decimal aleatorio` and `semilla aleatoria es`. The workspace also features a teal backpack icon in the top right and a green trash can icon in the bottom right.

Luego el evento del botón, con este menú y esta opción

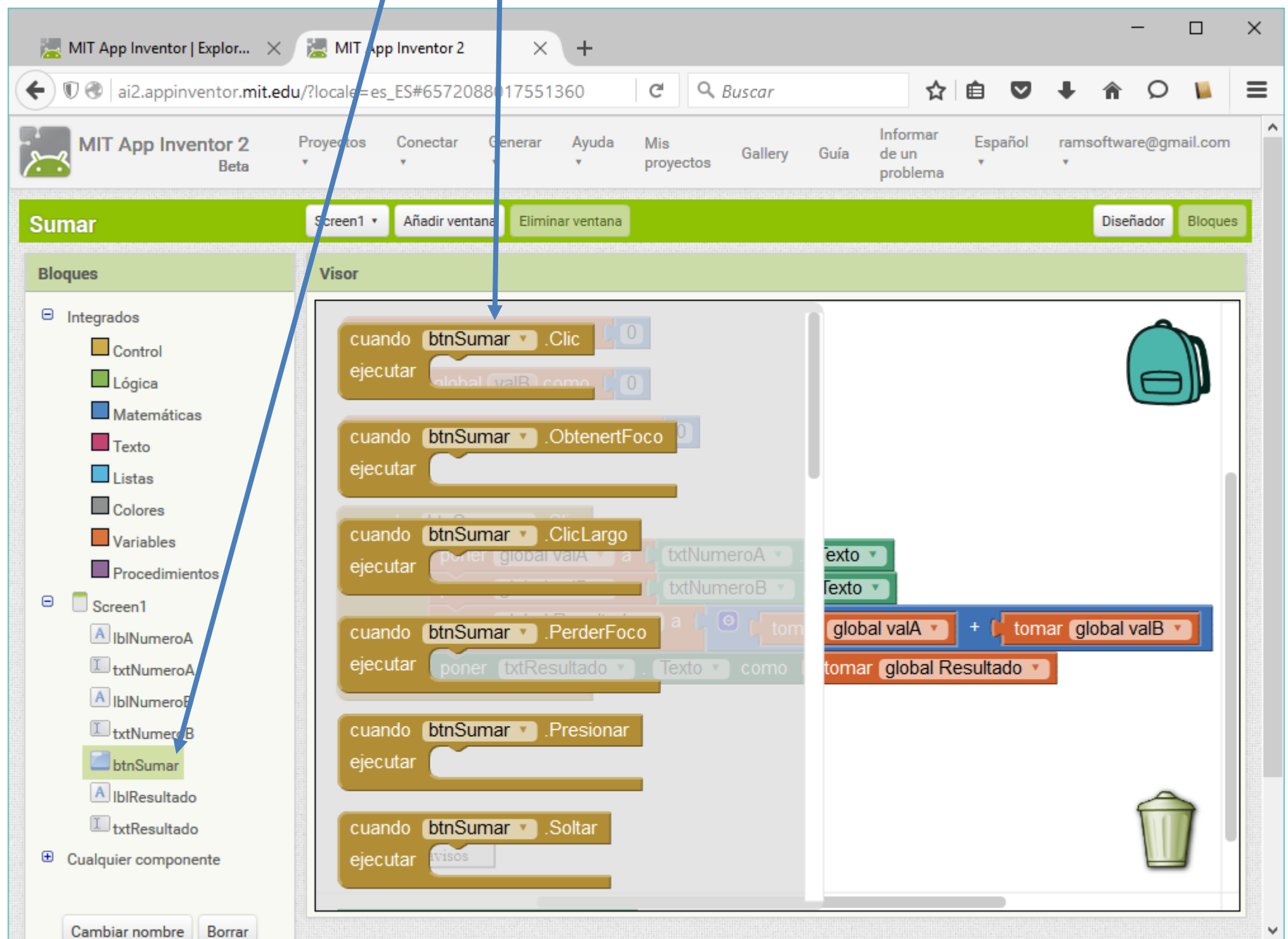


Imagen 59: Menú para el evento del botón

Y se prueba la aplicación



Imagen 60: El programa al iniciar el emulador



Imagen 61: Similar a un Smartphone, se requiere el teclado virtual en pantalla



Imagen 62: Funcionamiento correcto

Paso 8. Haciendo operaciones matemáticas más complejas

Tenemos una expresión algebraica como esta:

$$x = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

¿Cómo se puede implementar en App Inventor?

Para iniciar, debemos cambiar para que se vean explícitamente las operaciones matemáticas

$$x = \frac{-b + \sqrt{b * b - 4 * a * c}}{2 * a}$$

Luego lo pasamos a formato horizontal y quitamos ese menos inicial agregando un cero, la expresión sigue siendo equivalente

$$x = (0 - b + \text{raiz}(b * b - 4 * a * c)) / (2 * a)$$

Definimos variables



Se capturan los valores



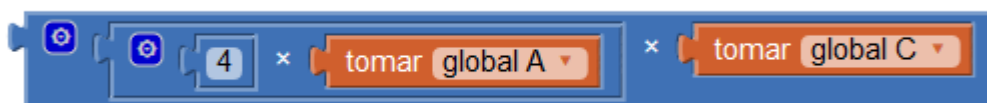
Tenemos la parte $(2 * a)$



Tenemos la parte $b * b$



Tenemos la parte $4 * a * c$ (se requieren dos piezas de multiplicación)



Tenemos la parte de $b * b - 4 * a * c$



Tenemos la parte de $\text{raiz}(b * b - 4 * a * c)$



Tenemos la parte de $(0 - b + \text{raiz}(b * b - 4 * a * c))$



Tenemos la parte de $(0 - b + \text{raiz}(b * b - 4 * a * c)) / (2 * a)$



Y esta es la aplicación

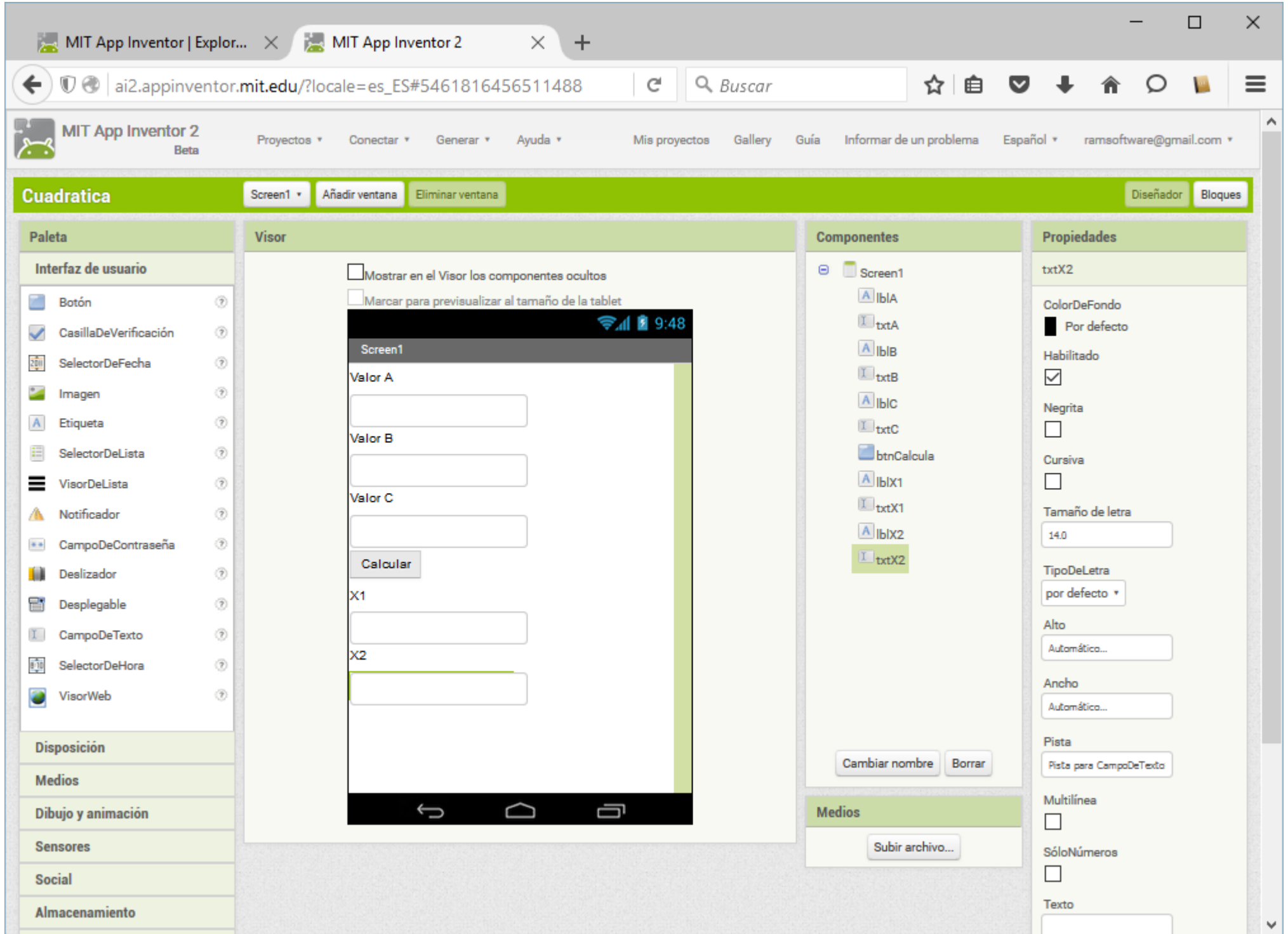
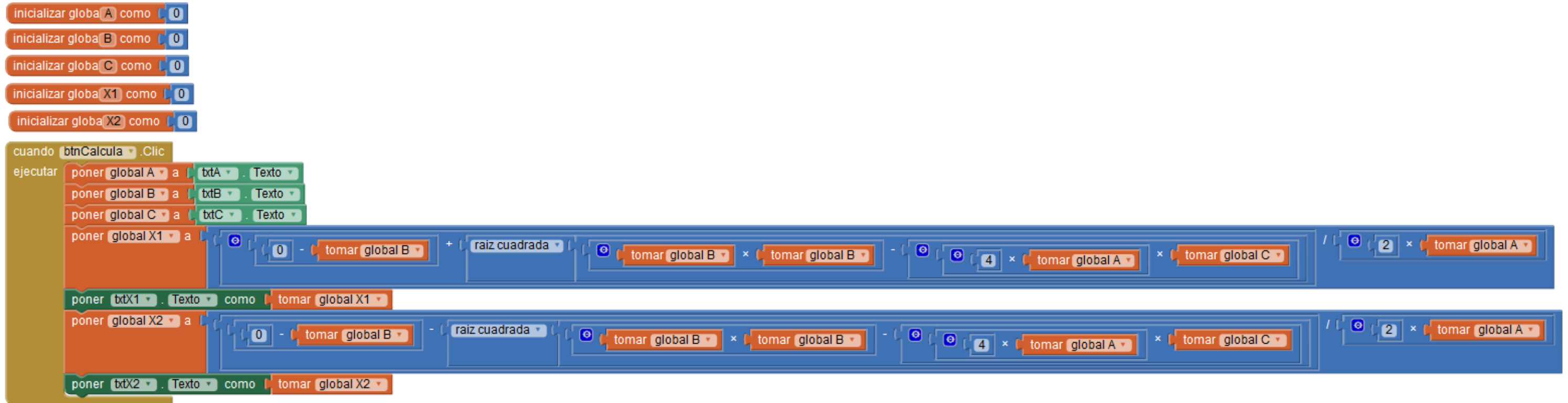


Imagen 63: Diseño de la aplicación

Este es el diagrama de bloques



Esta es su ejecución

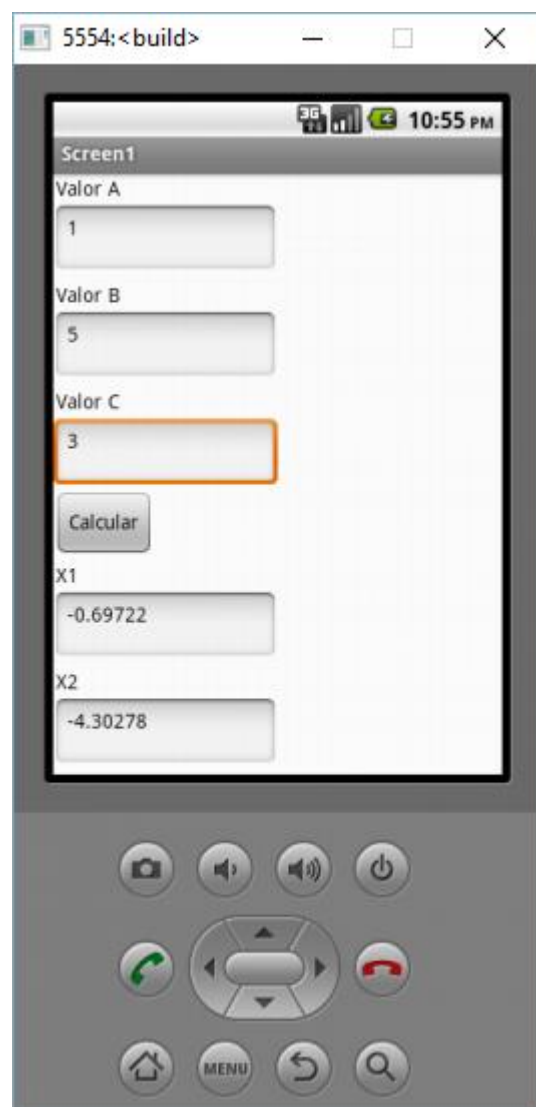


Imagen 64: Ejecución de la aplicación en el emulador

Y probado con Microsoft Excel

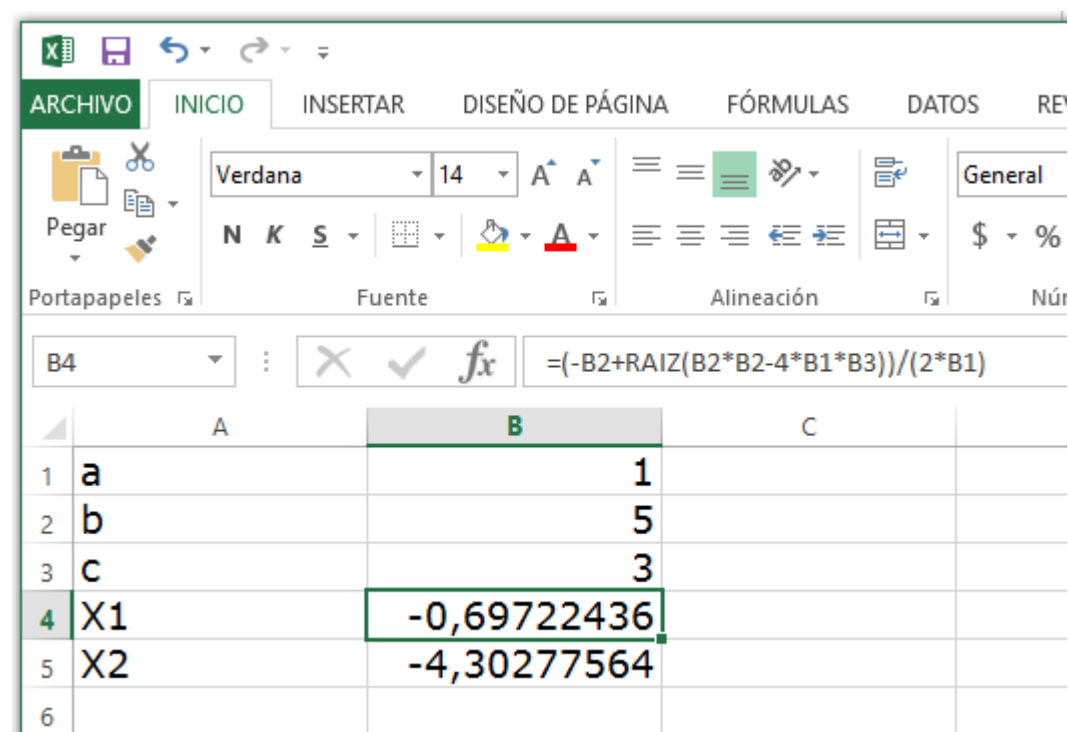


Imagen 65: Prueba de la operación en Excel

Es un poco complejo, ¿alguna forma de hacer más sencilla esa operación matemática? Si, una forma es crear variables e ir dividiendo la expresión en partes más pequeñas.

Paso 9. El SI condicional

Un algoritmo sencillo de si condicional

```
Si Edad >= 18 entonces
    Imprima "Es mayor de edad"
De lo contrario
    Imprima "Es menor de edad"
```

Implementándolo en App Inventor:

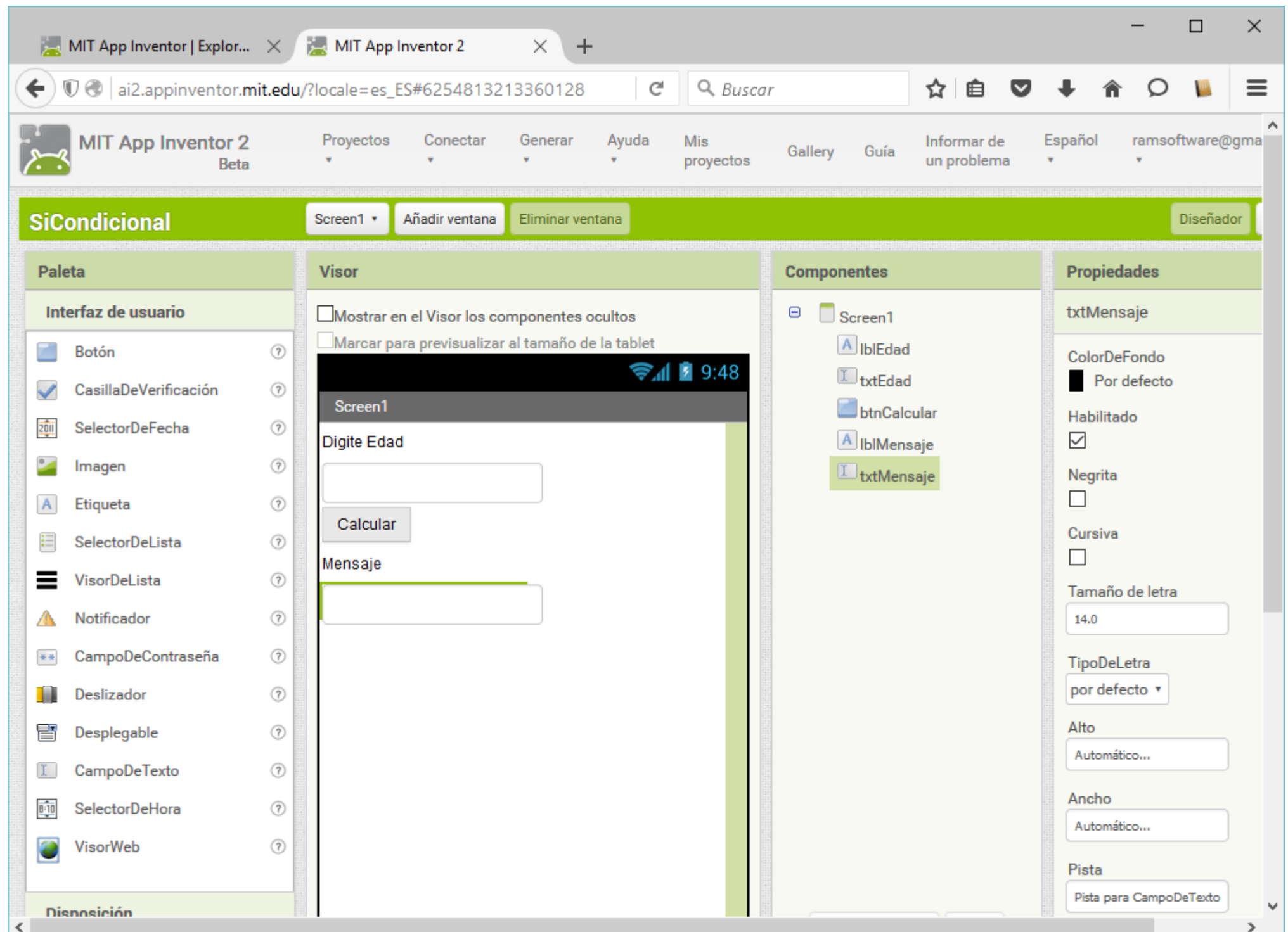


Imagen 66: Diseño de pantalla que pide una edad en la primera caja de texto y luego muestra si es menor o mayor de edad

Este es el algoritmo en el formato de App Inventor

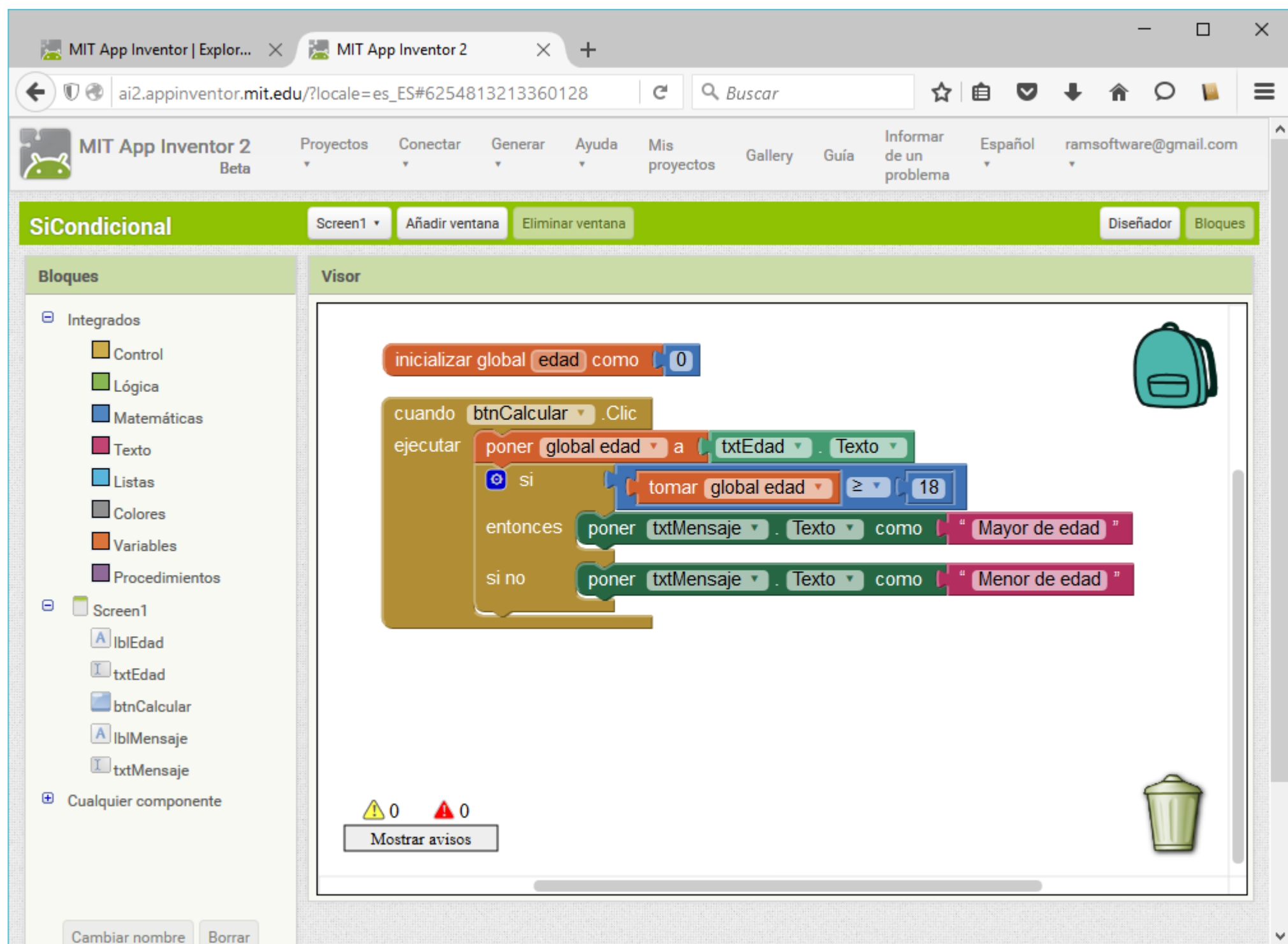


Imagen 67: Algoritmo en App Inventor que implementa el si condicional

La pieza "si entonces" se encuentra en "Control" y esta es la que se usa

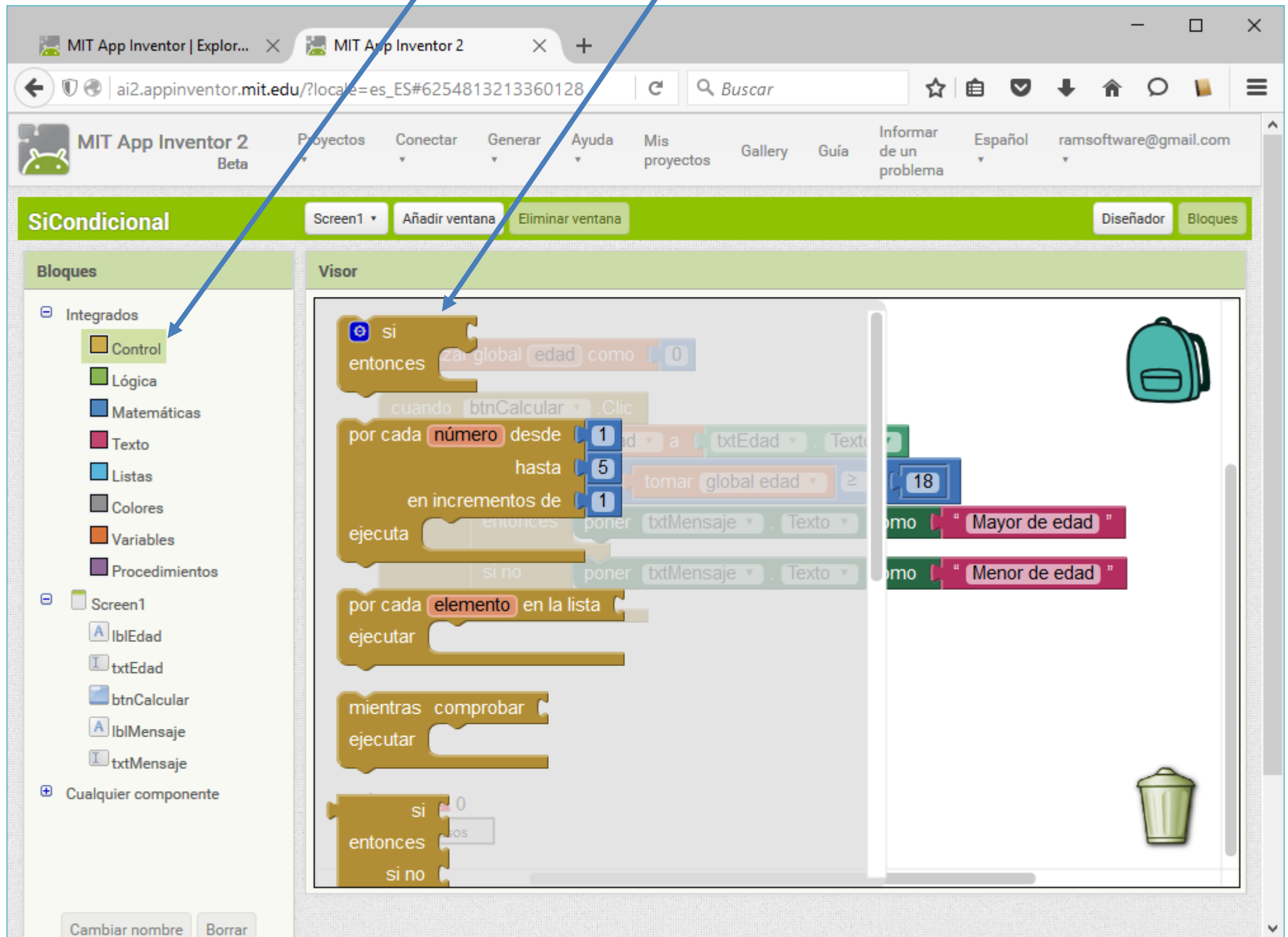


Imagen 68: La pieza del sí condicional se encuentra en Control

¡OJO! para hacer uso del "si no" o "de lo contrario" hay que dar clic aquí y arrastrar la figura

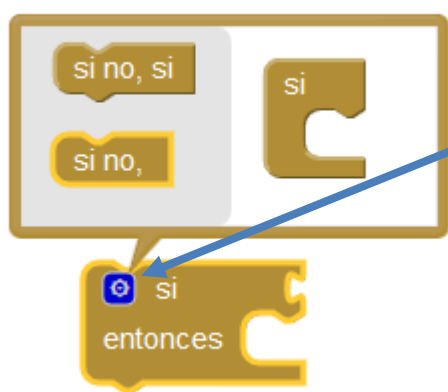


Imagen 69: De clic en el botón azul para obtener más opciones de la ficha

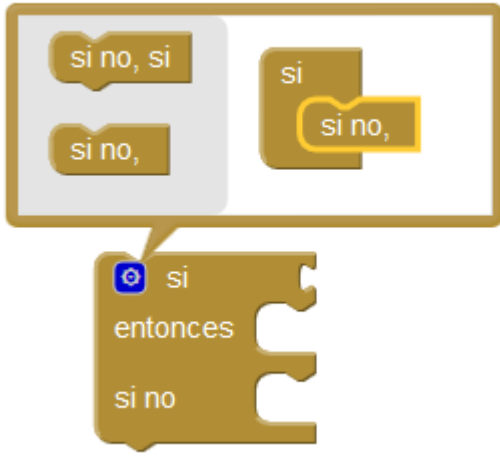


Imagen 70: Agregue el "si no," para que la ficha tenga un "de lo contrario"

En "Matemáticas" están los comparativos

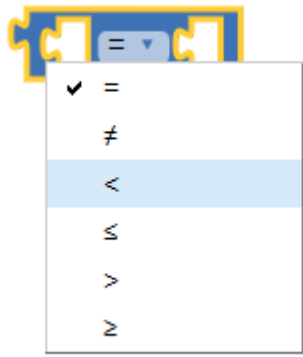
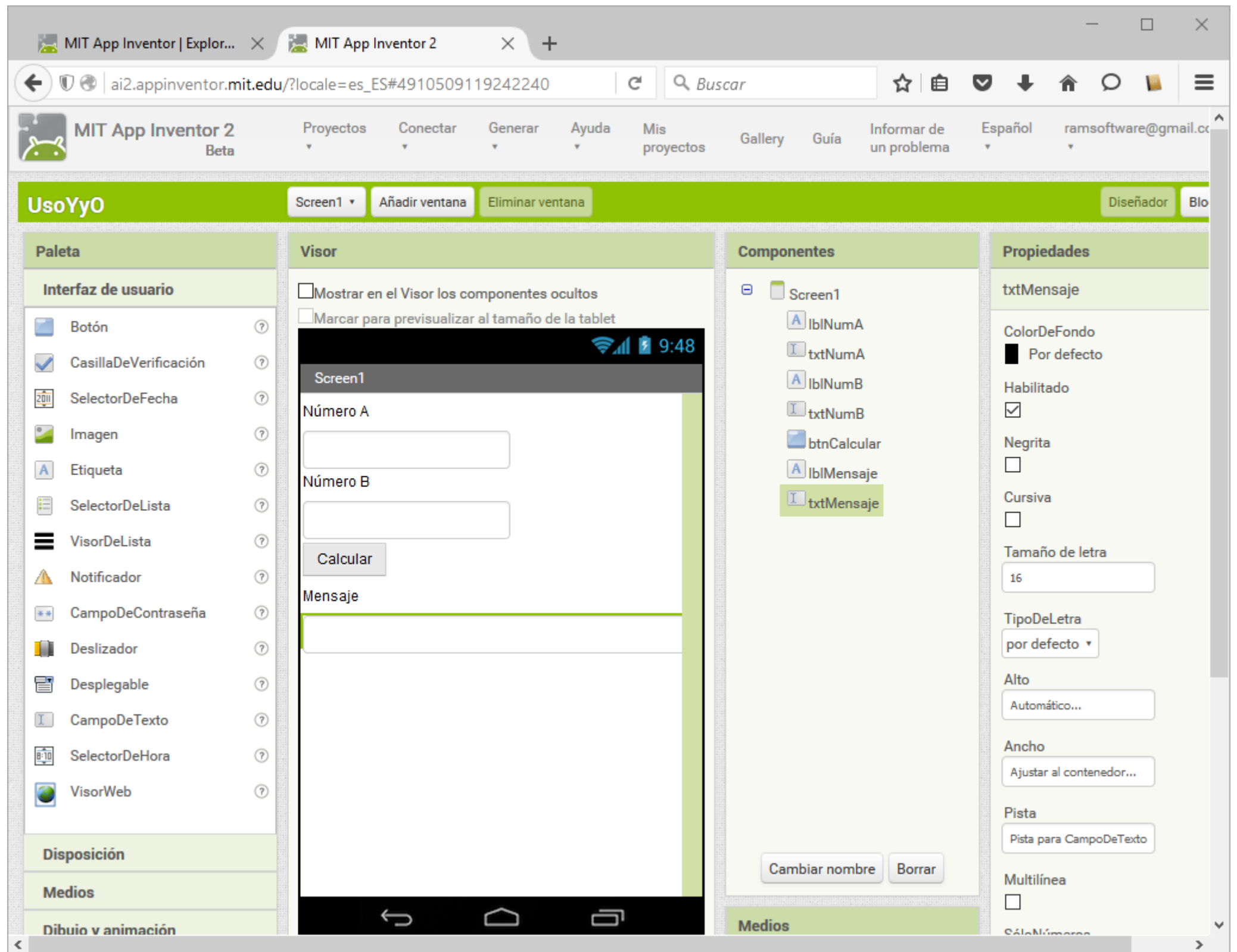


Imagen 71: Operadores de comparación en "Matemáticas"

Uso del Y (And) y el O (Or)

El siguiente ejemplo es un algoritmo que verifica si dos números ingresados son pares. Hace uso del Y.



Y este es el diagrama de bloques



Ejecutando

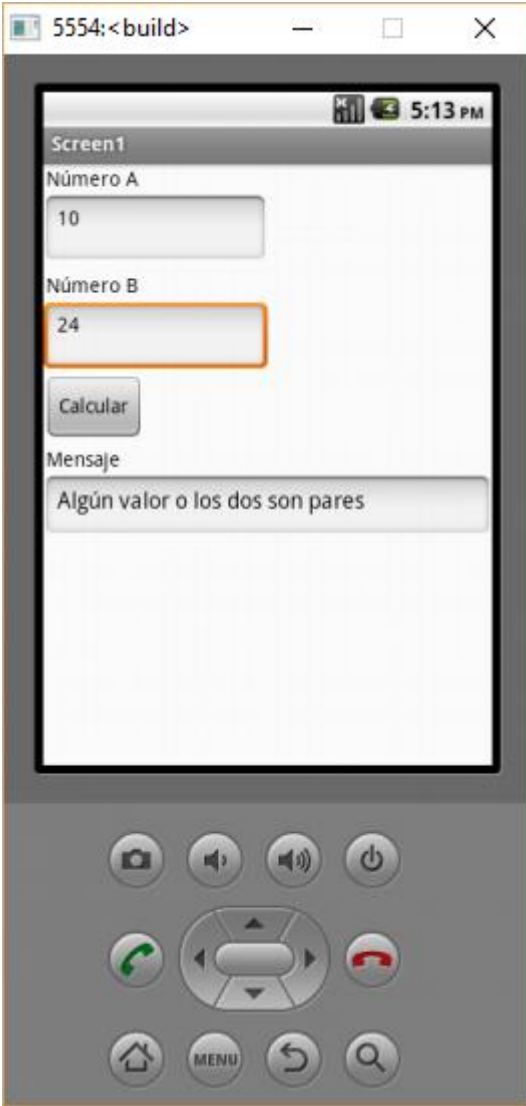


Cambiando para hacer uso del O (Or)



```
inicializar global A como 0  
inicializar global B como 0
```

```
cuando btnCalcular .Clic  
ejecutar  
  poner global A a txtNumA . Texto  
  poner global B a txtNumB . Texto  
  si  
    módulo de tomar global A ÷ 2 = 0  
    módulo de tomar global B ÷ 2 = 0  
  entonces poner txtMensaje . Texto como "Algún valor o los dos son pares"  
  si no poner txtMensaje . Texto como "Ningún valor es par"
```



Paso 10. Ciclos

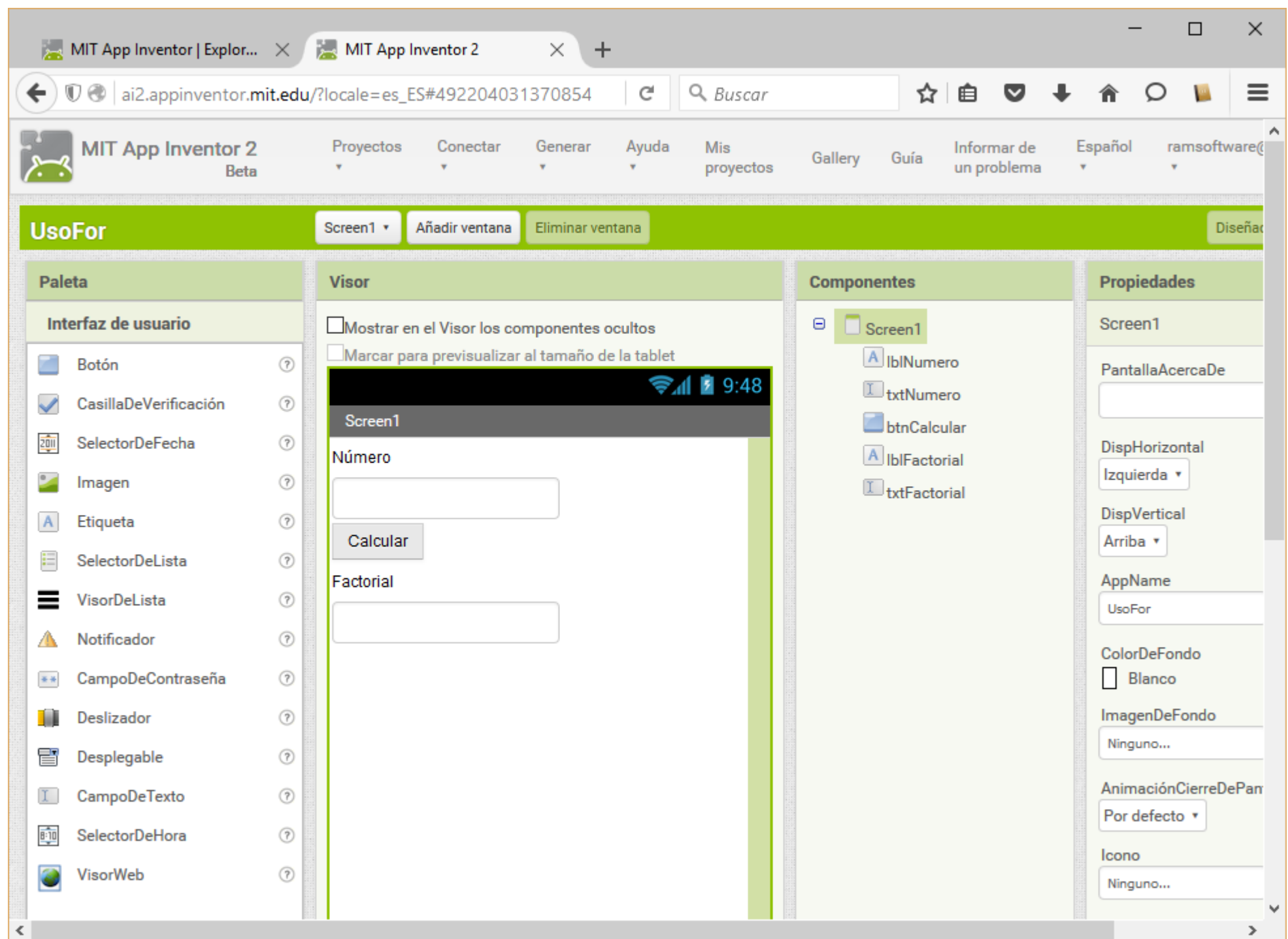
Un algoritmo que calcula el factorial de un número es así:

Algoritmo Factorial

```
Inicio
Leer Numero
Acum = 1
Desde cont = 1 hasta Numero paso 1 hacer
    Acum = Acum * cont
Fin Desde
Imprima Acum
```

Fin

En App Inventor se implementa así:



inicializar global **numero** como 0

inicializar global **acum** como 1

```
cuando btnCalcular .Clic
ejecutar
  poner global numero a txtNumero . Texto
  poner global acum a 1
  por cada cont desde 1
    hasta tomar global numero
    en incrementos de 1
    ejecuta poner global acum a tomar global acum * tomar cont
  poner txtFactorial . Texto como tomar global acum
```

La pieza "Desde" se encuentra en "Control":

The screenshot shows the MIT App Inventor 2 web interface. The browser address bar displays `ai2.appinventor.mit.edu/?locale=es_ES#4922040313708544`. The page title is "MIT App Inventor 2 Beta". The main workspace is titled "UsoFor" and shows a Scratch-style block editor. The left sidebar contains a "Bloques" (Blocks) palette with categories: Integrados (Control, Lógica, Matemáticas, Texto, Listas, Colores, Variables, Procedimientos), Screen1 (lblNumero, txtNumero, btnCalcular, lblFactorial, txtFactorial), and "Cualquier componente". The main editor area shows the following code blocks:

- When green flag clicked: `numero como 0`
- if-then block: `si` (true) `entonces` `global acum como 1`
- for loop: `por cada número desde 1 hasta 5` in increments of 1, `ejecuta` `por cada cont desde 1 hasta tomar global numero`
- for each loop: `por cada elemento en la lista` `ejecuta` `ejecuta poner global acum a tomar global acum × tomar cont`
- while loop: `mientras comprobar` `ejecuta` `comprobar` `comprobar` `como` `tomar global acum`
- if-then-else block: `si` `entonces` `si no` `0` `Mostrar avisos`
- Final block: `ejecutar`

The screenshot shows a mobile device emulator window titled "5554:<build>". The application interface displays a screen with the following elements:

- Header: "Screen1"
- Input field: "Número" with the value "4" entered.
- Button: "Calcular"
- Output field: "Factorial" with the value "24" displayed.
- Keyboard: A virtual QWERTY keyboard is visible below the screen.
- Home screen: A virtual home screen with various icons (camera, volume, power, home, menu, search) is visible at the bottom.

Paso 11. Operaciones con cadenas (Básico)

A. Longitud de cadenas

Determinar el tamaño de una cadena incluyendo los espacios

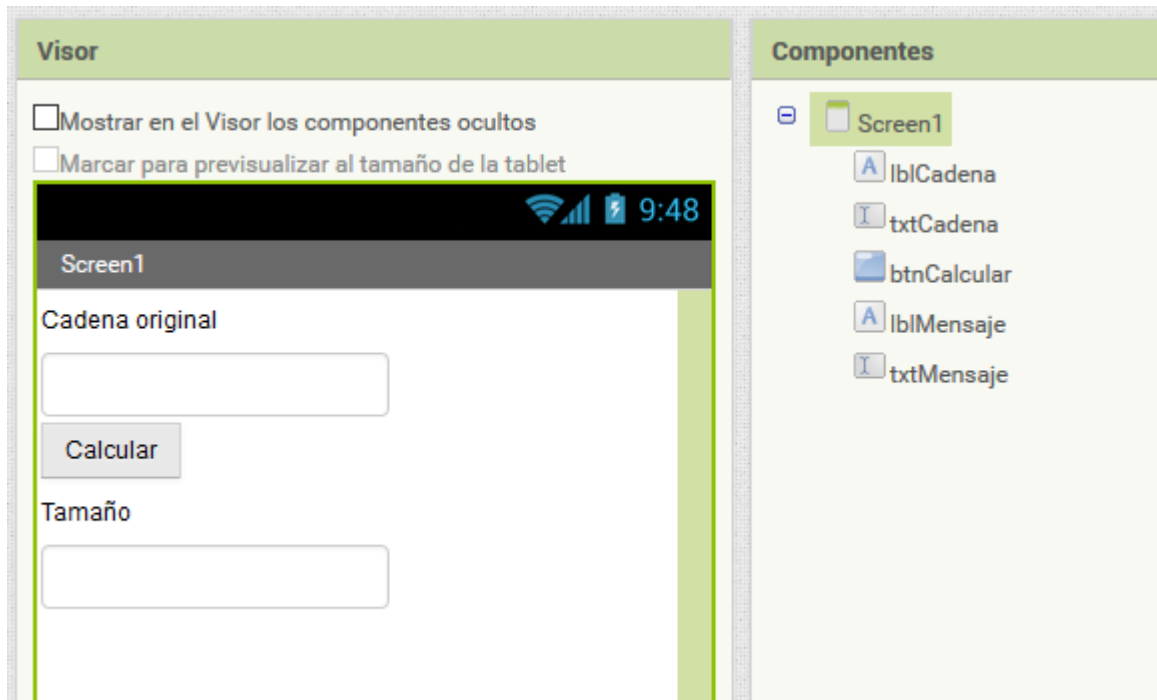


Imagen 72: Diseño de la pantalla para mostrar la funcionalidad tamaño de texto



Imagen 73: Se utiliza el bloque "Longitud" de "Texto"

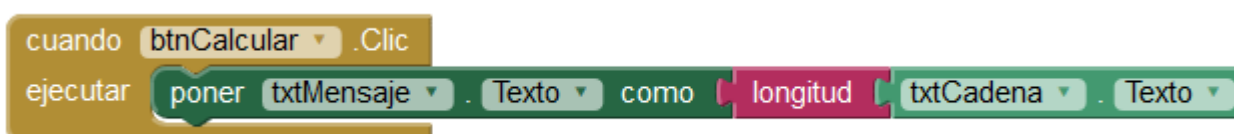


Imagen 74: Uso de la pieza "call length text"

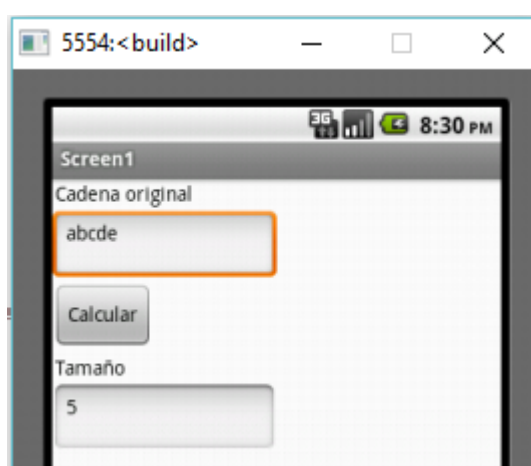


Imagen 75: Ejecución del programa

B. Chequear si una cadena está vacía

Chequear si una cadena no tiene ni un solo caracter

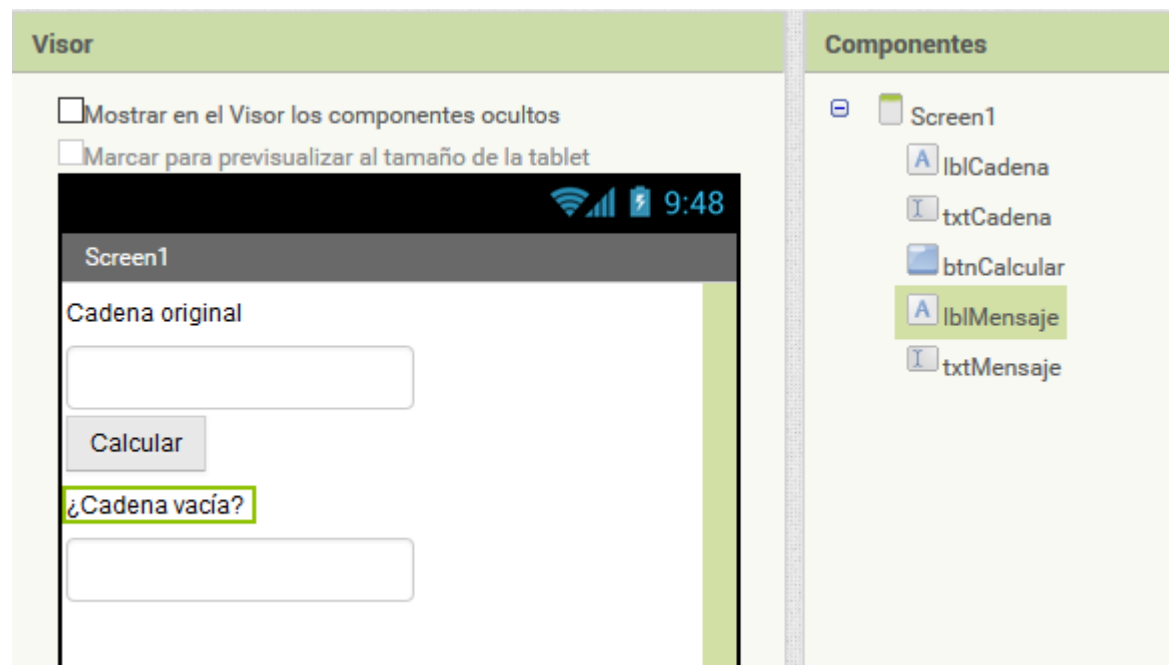


Imagen 76: Diseño de la pantalla para mostrar la funcionalidad chequear cadena vacía

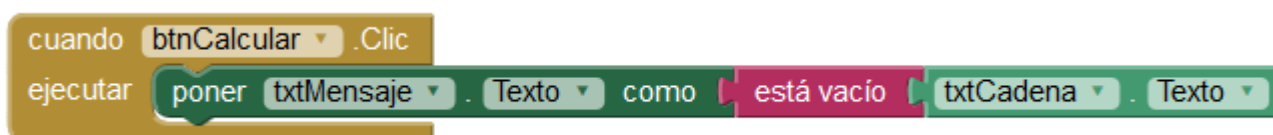


Imagen 77: Algoritmo para probar si la cadena es vacía

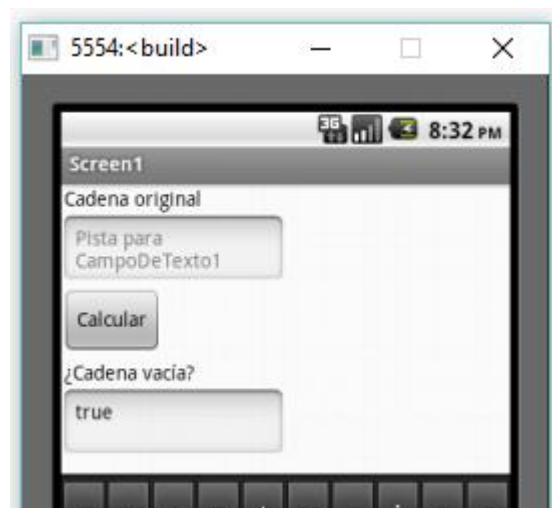


Imagen 78: Observe que se retorna un valor booleano

C. Quitar espacios al inicio y al final

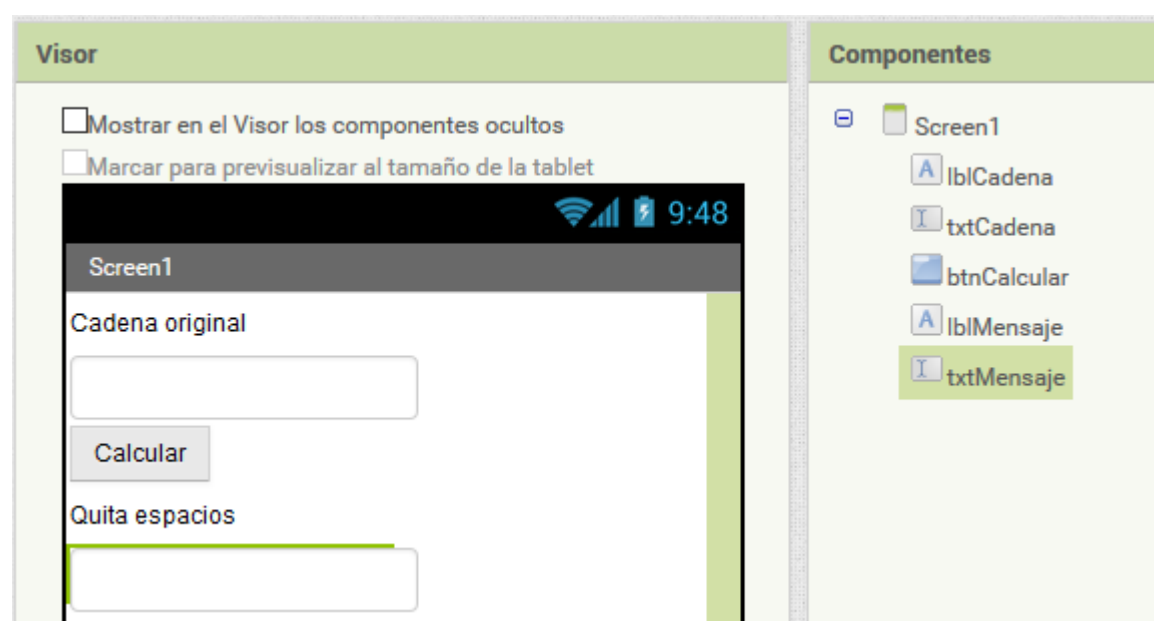


Imagen 79: Diseño de la aplicación



Imagen 80: El bloque "trim" quita los espacios al inicio y final

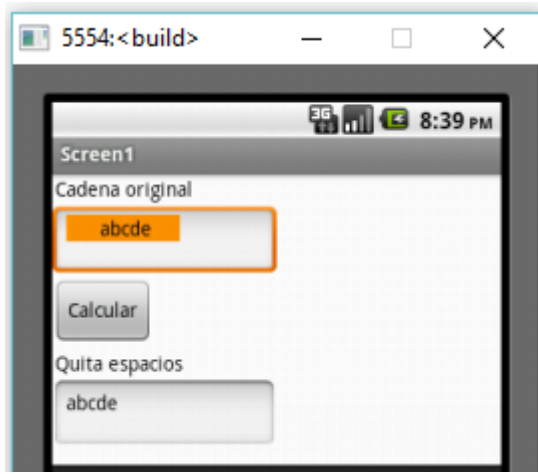


Imagen 81: Ejemplo de ejecución

D. Convertir a mayúsculas

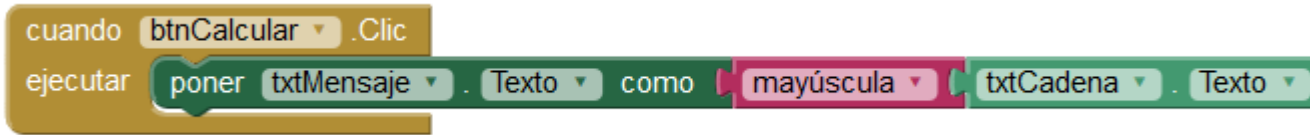


Imagen 82: El bloque "mayúscula" convierte a mayúsculas

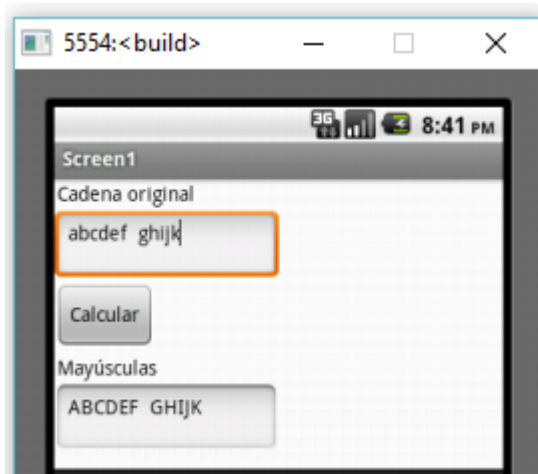


Imagen 83: Ejemplo de ejecución

E. Convertir a minúsculas

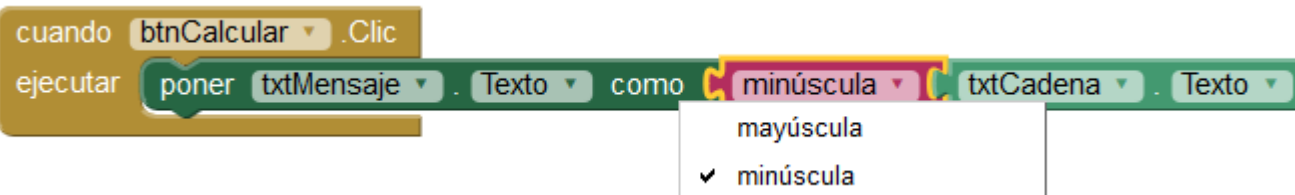


Imagen 84: El bloque "minúscula" convierte a minúsculas

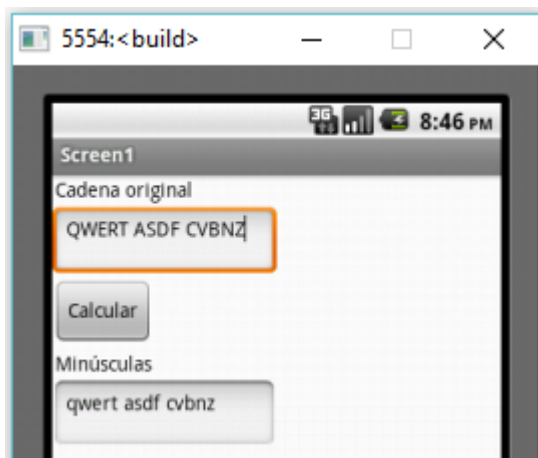


Imagen 85: Convertir a minúsculas

F. Reemplazar una subcadena por otra

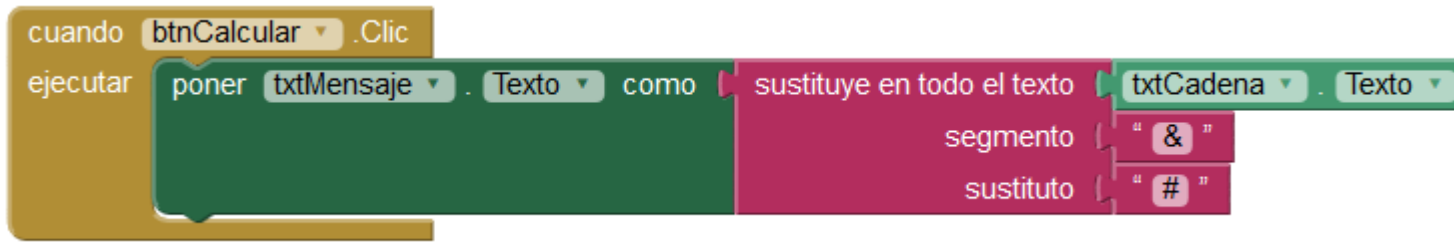


Imagen 86: Reemplaza el carácter & por el carácter #

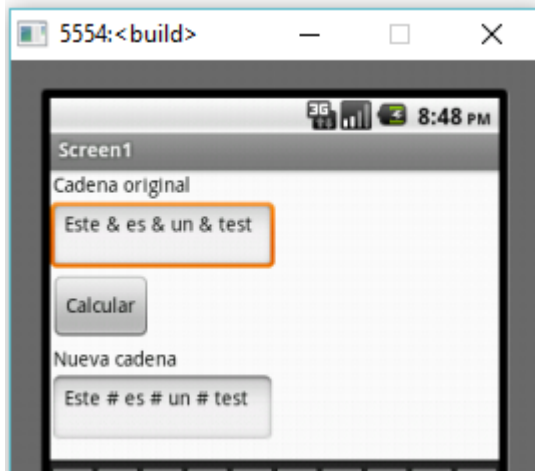


Imagen 87: Ejemplo de ejecución

H. Concatenación de cadenas

Suma varias cadenas y muestra el resultado

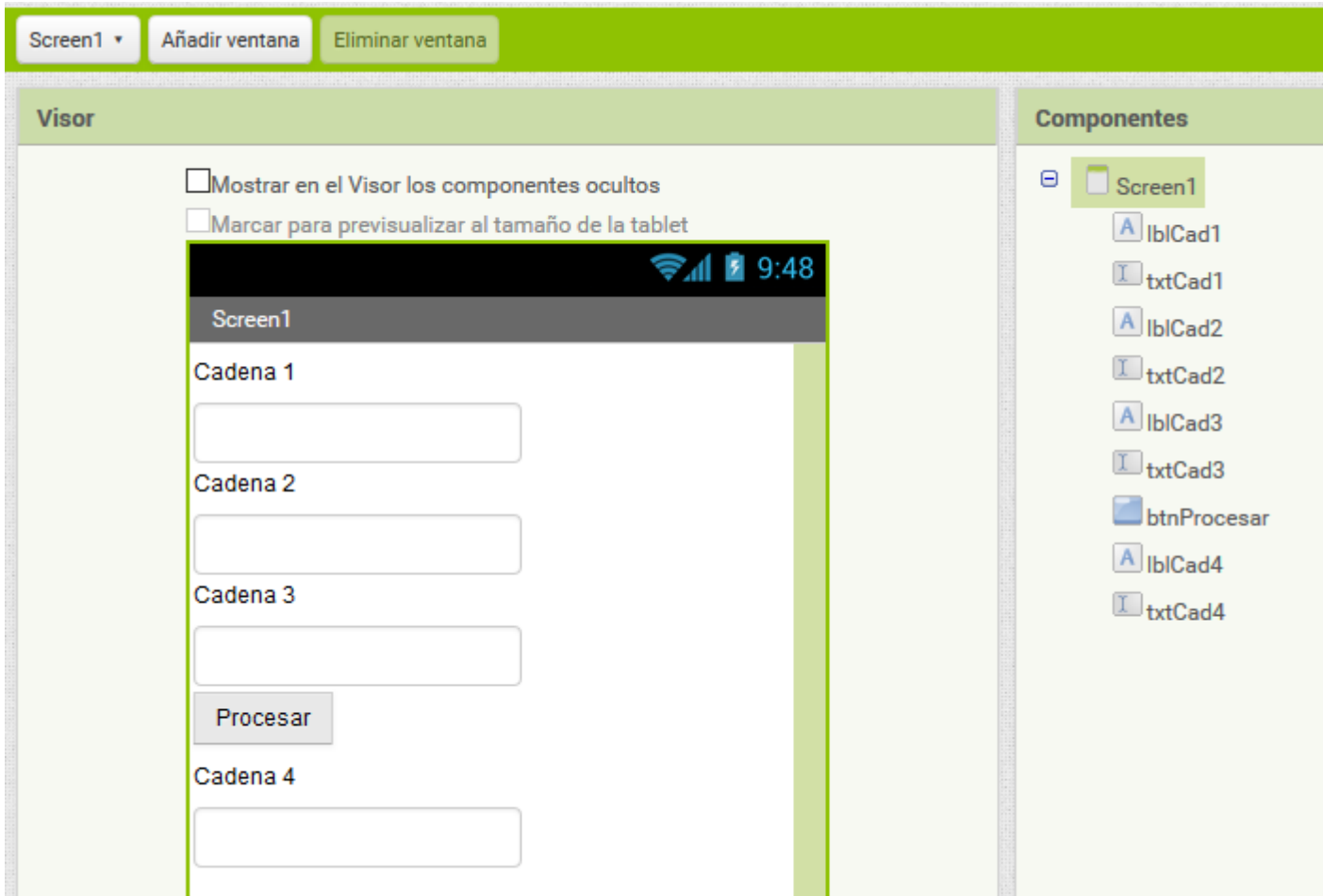


Imagen 88: Diseño para probar la funcionalidad de concatenar cadenas

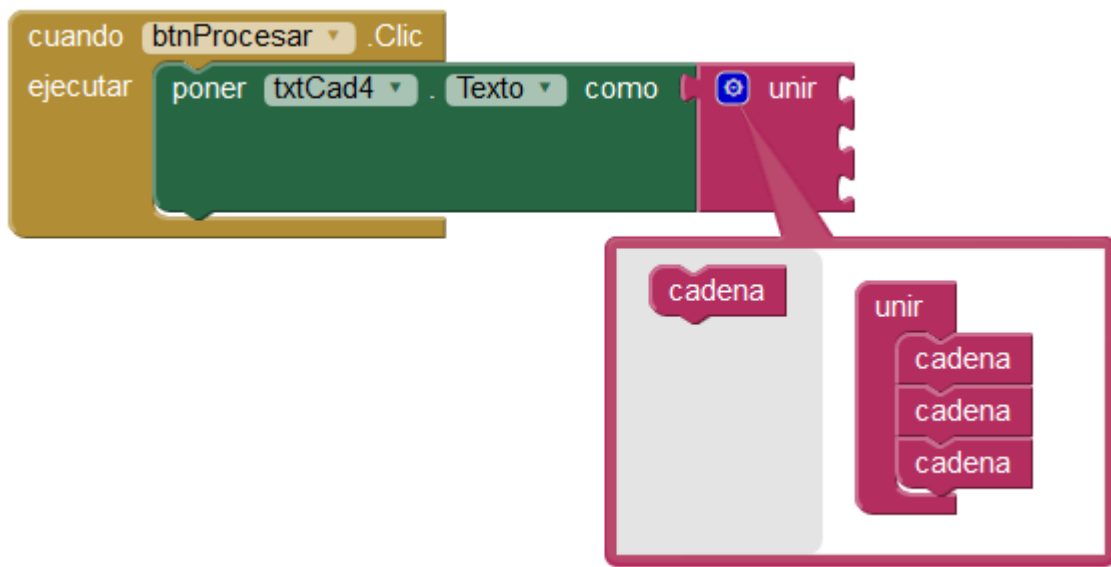


Imagen 89: Modificando el bloque "unir" agregando más cadenas

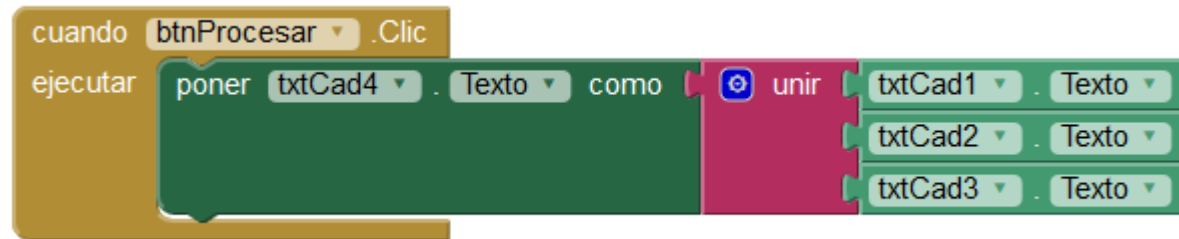


Imagen 90: Algoritmo completo

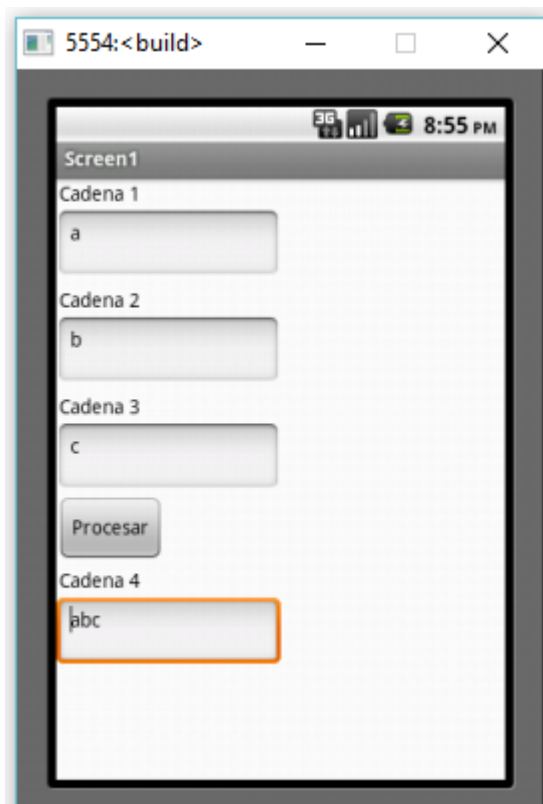


Imagen 91: Ejecución del algoritmo de conectar cadenas

I. Comparación de cadenas

Compara los caracteres de cada cadena y retorna verdadero(true) si son iguales, falso(false) en caso contrario

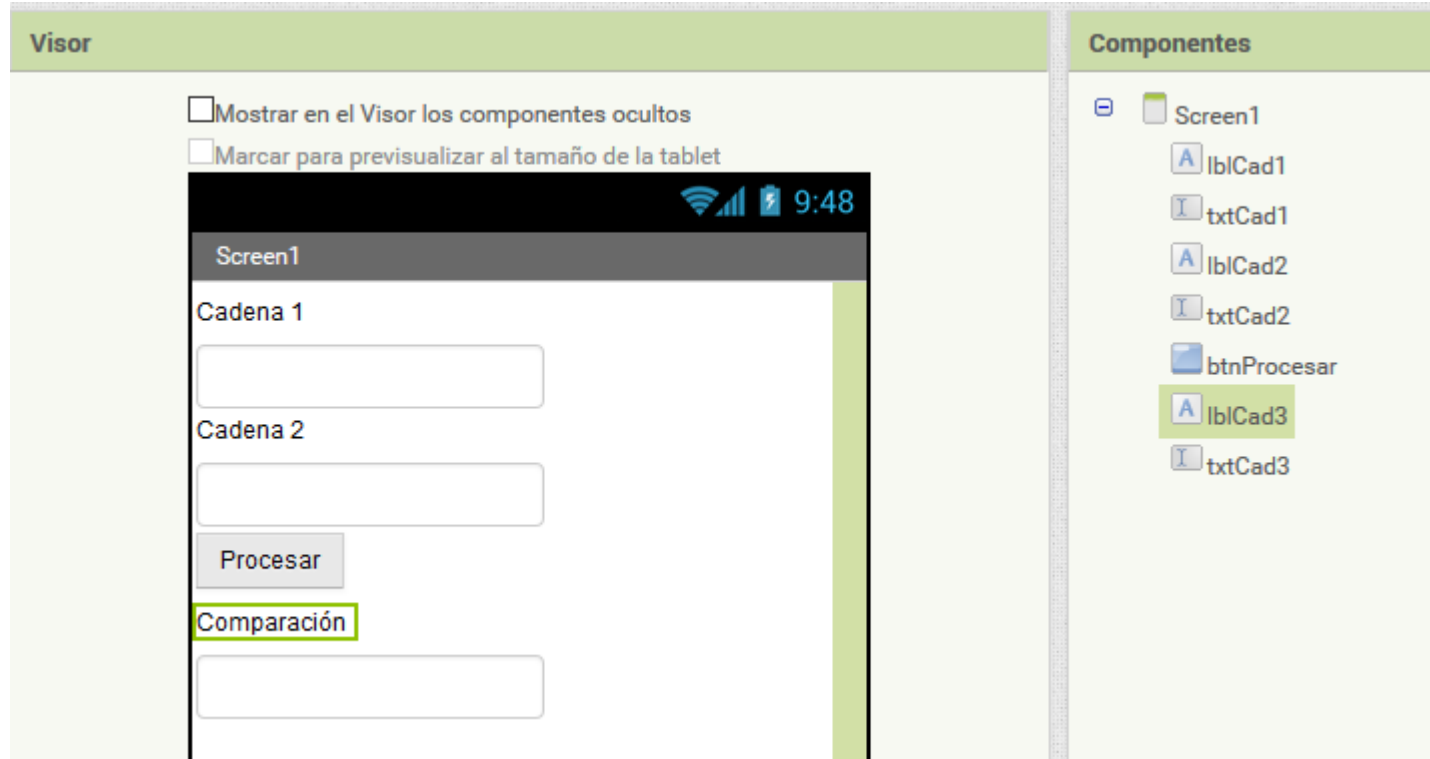


Imagen 92: Diseño de pantalla para probar la funcionalidad de comparación de cadenas

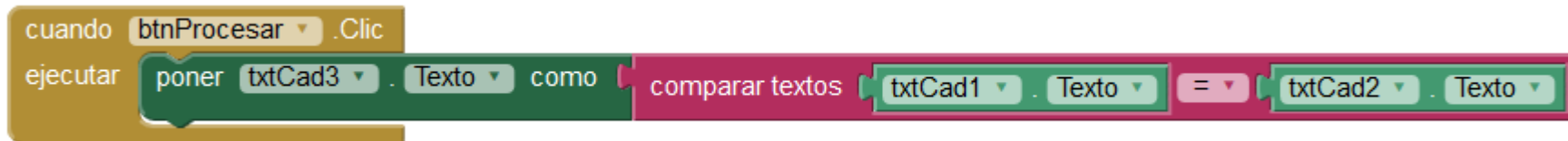


Imagen 93: Algoritmo de comparación de cadenas

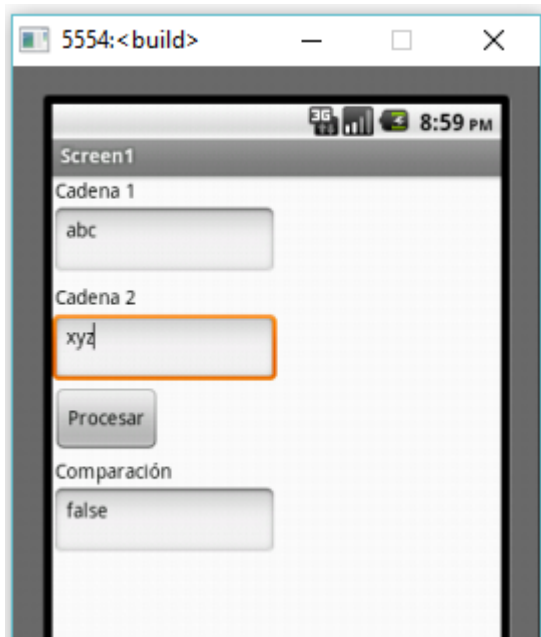


Imagen 94: Ejemplo de ejecución de comparación de cadenas. Resultado booleano.

J. Señala en qué posición se encuentra una determinada subcadena en la cadena



Imagen 95: Diseño GUI de la pantalla para probar la búsqueda de subcadenas

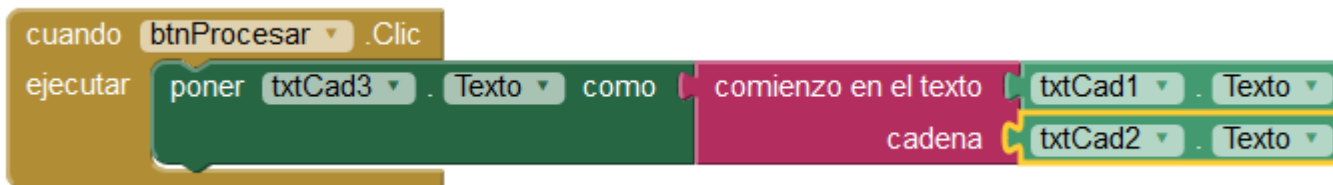


Imagen 96: El bloque "comienzo en el texto" busca una cadena dentro de otra y retorna la posición de encuentro.

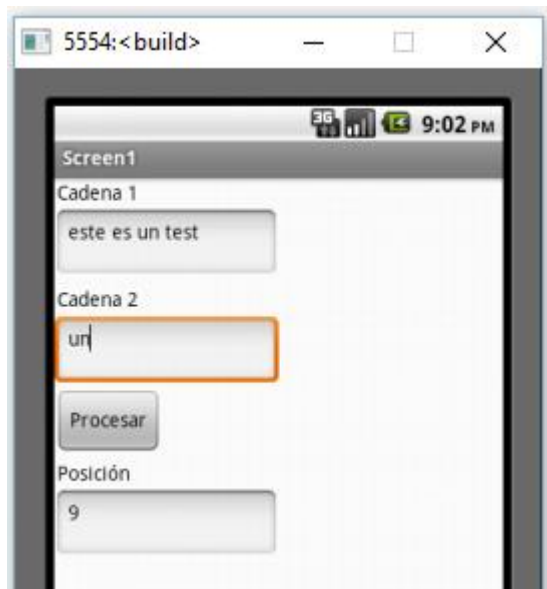
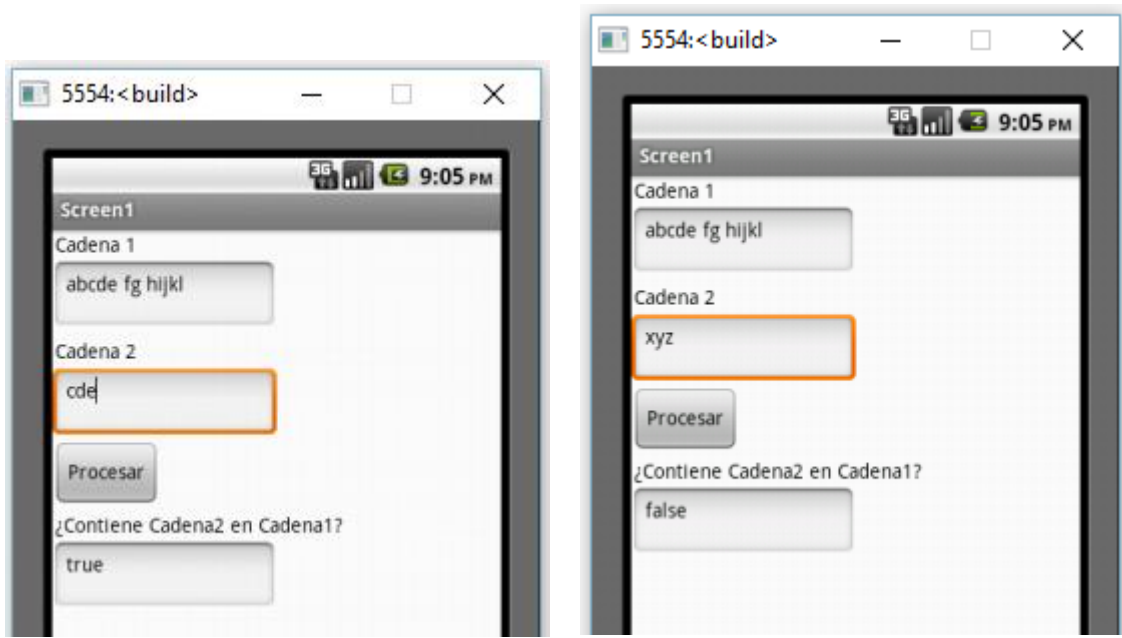


Imagen 97: Retorno de la posición de la subcadena en una cadena

K. Retorna verdadero si encuentra la subcadena dentro de la cadena



Imagen 98: El bloque "contiene texto" retorna verdadero si encuentra una subcadena dentro de una cadena



L. Extraer una subcadena

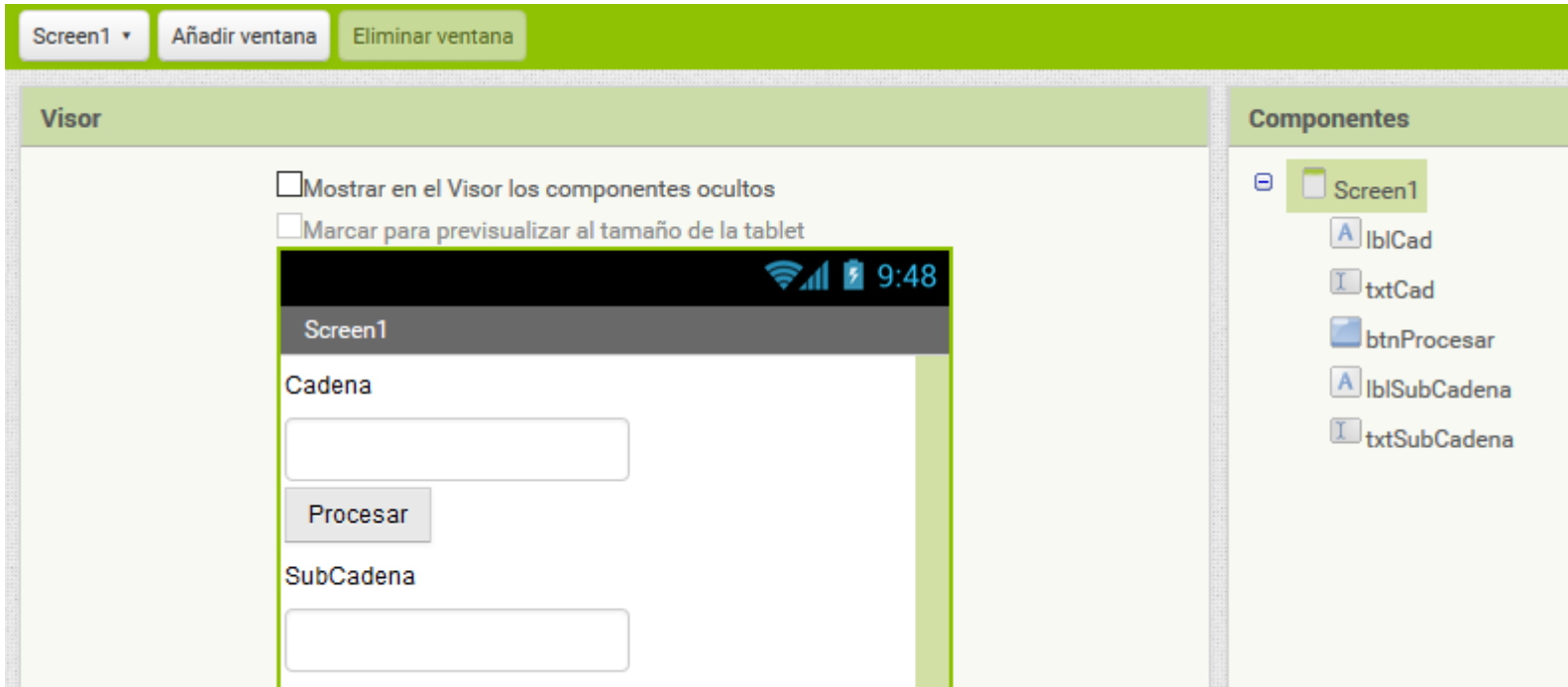
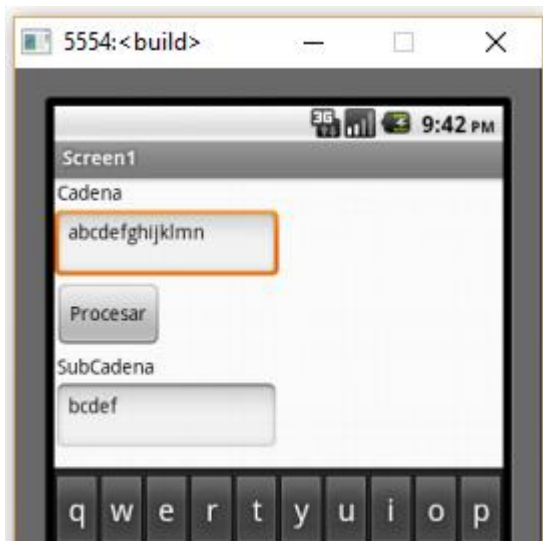


Imagen 99: Extrae una subcadena de la cadena original desde la primera letra y toma cuatro(4) caracteres



Paso 12. Operaciones con listas

Una lista es una colección de ítems



Imagen 100: Diseño de pantalla para el trabajo con listas

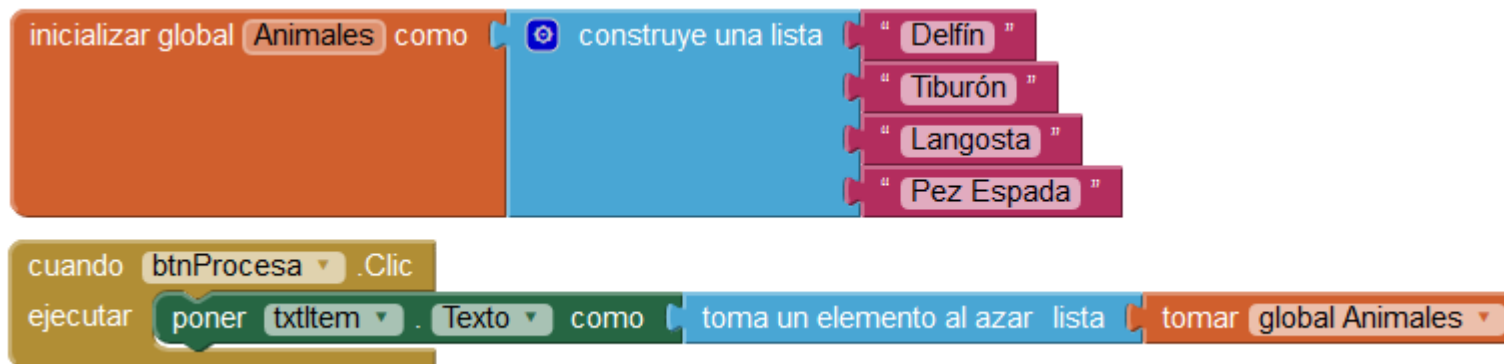
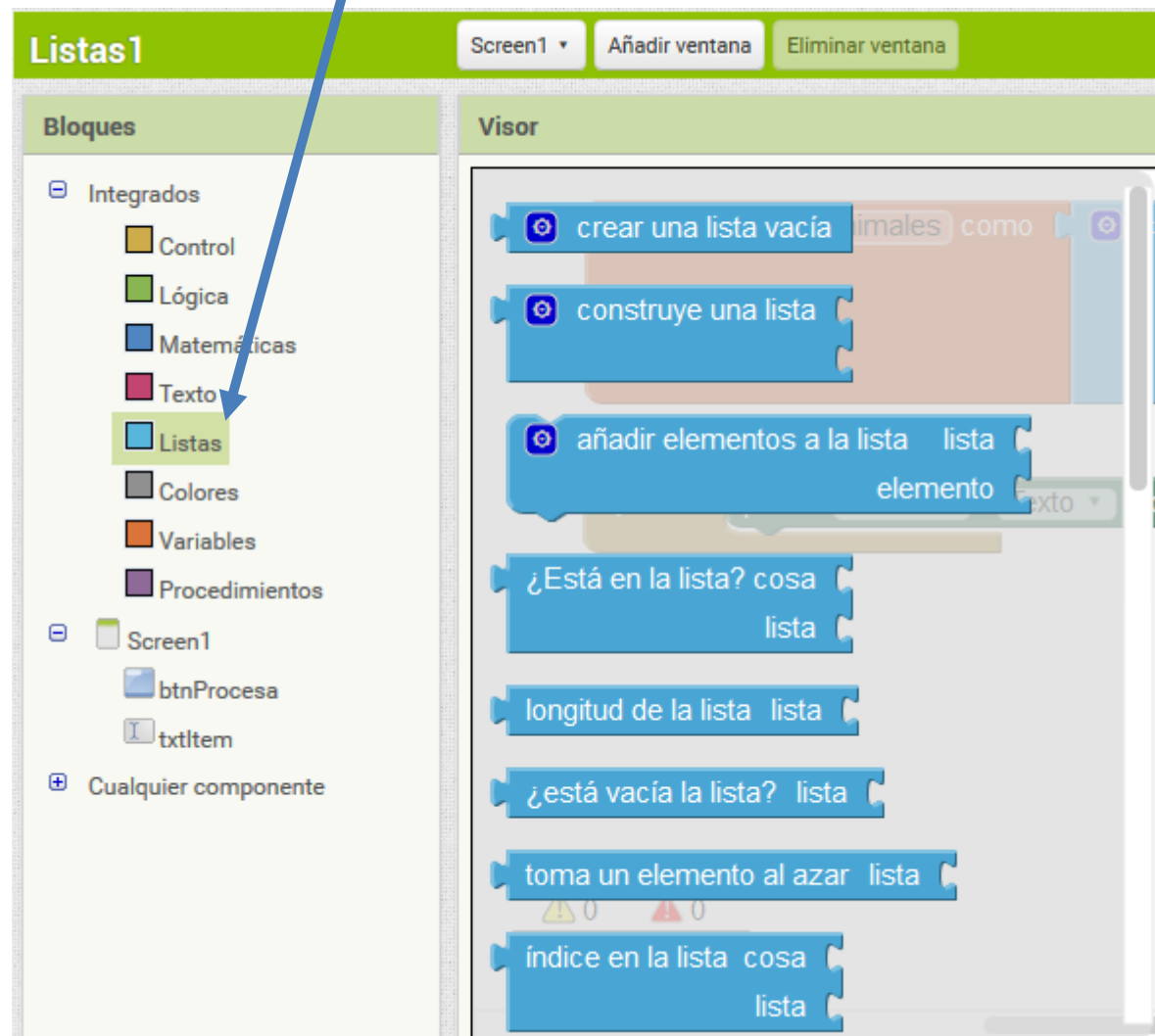
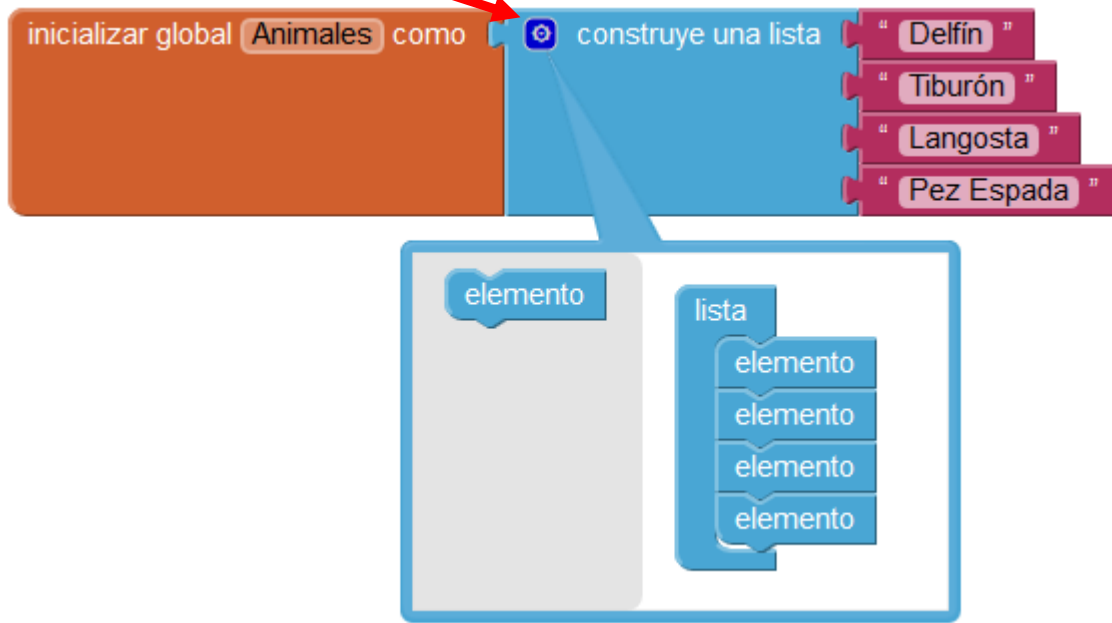


Imagen 101: Primero se define una lista y cuando se oprima el botón se selecciona al azar un ítem de esa lista

Se toma de la opción lista



Y se hace uso del menú de la pieza para agregar más ítems a la lista.



A. Longitud de una lista

Muestra cuantos ítems tiene una lista

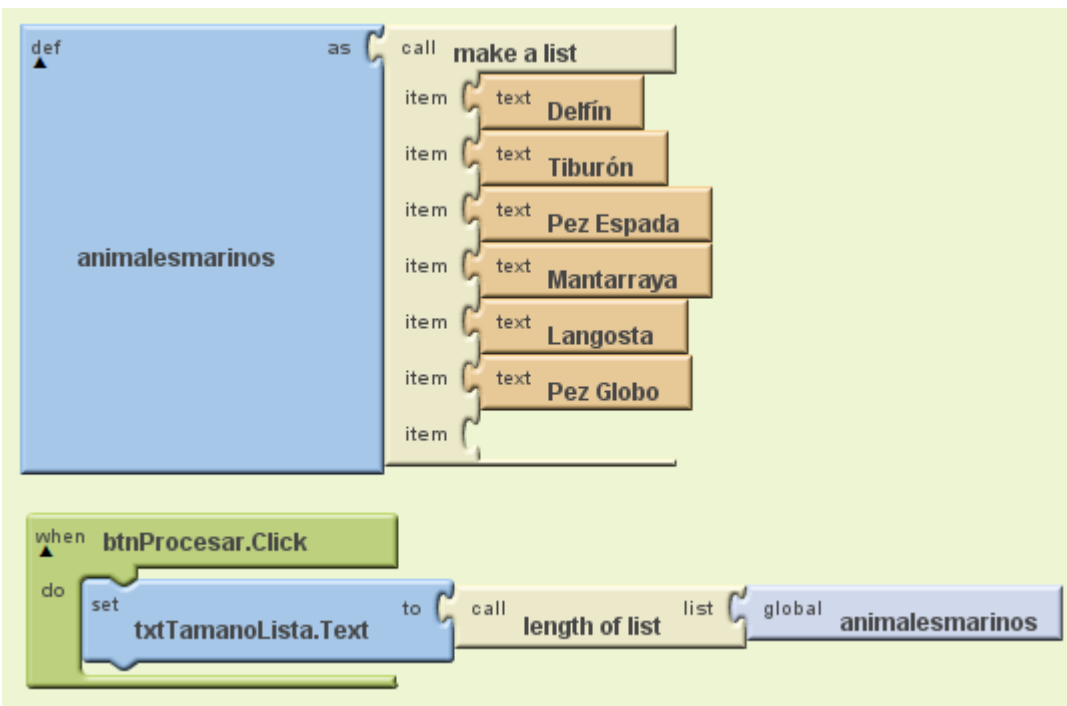


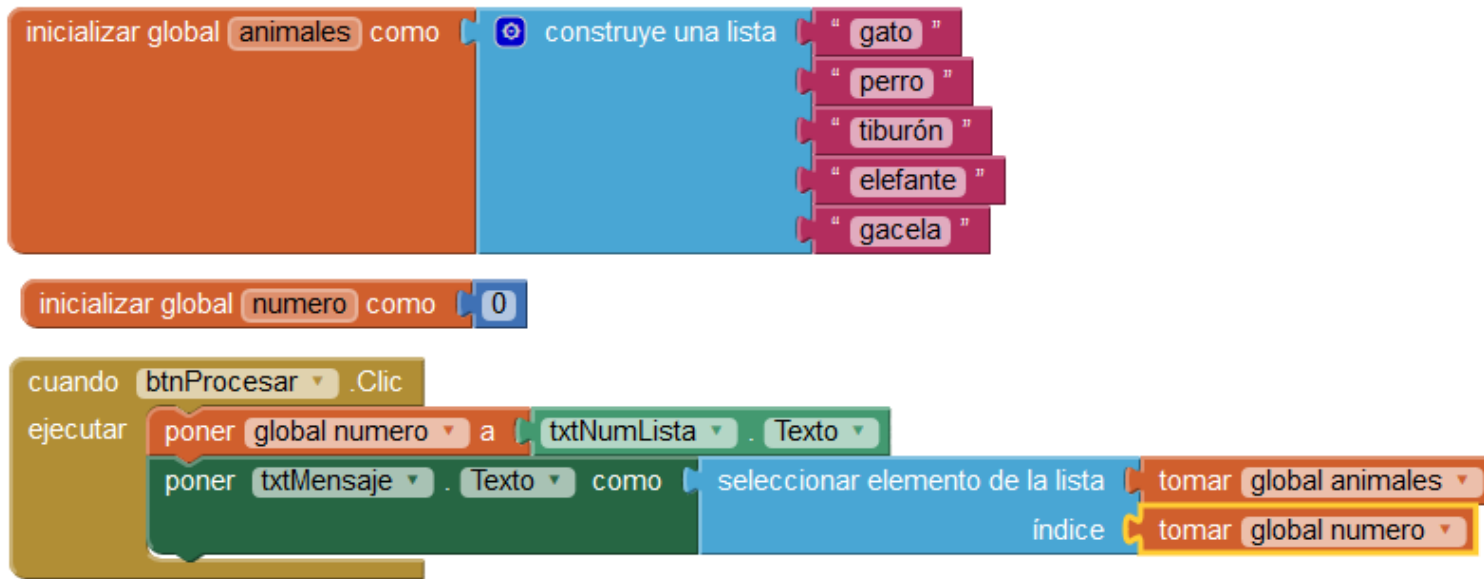
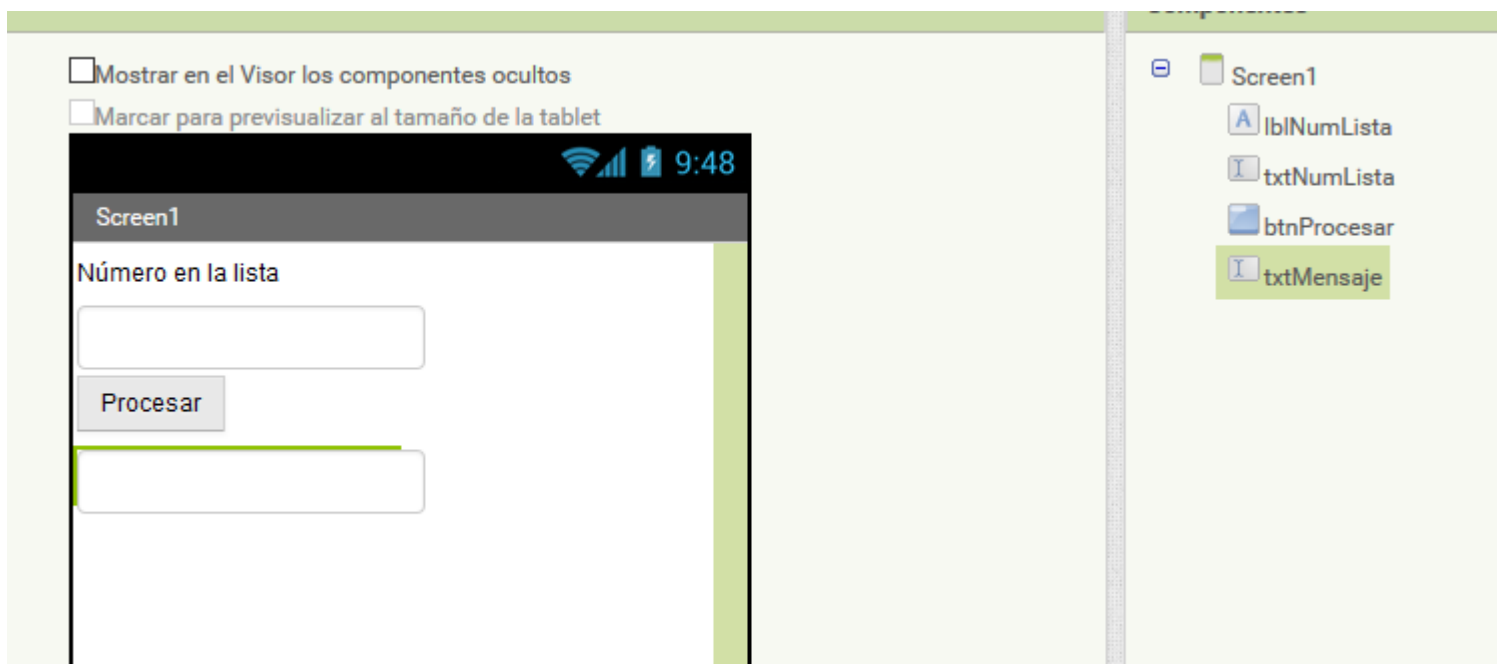
Imagen 102: Tamaño de la lista



Imagen 103: Ejecución tamaño de lista

B. Seleccionar un ítem de una lista

Dada una lista de ítems, poder seleccionar uno en particular y mostrarlo en un campo de texto



C. Mostrar una lista en un "campo de texto multi-línea"

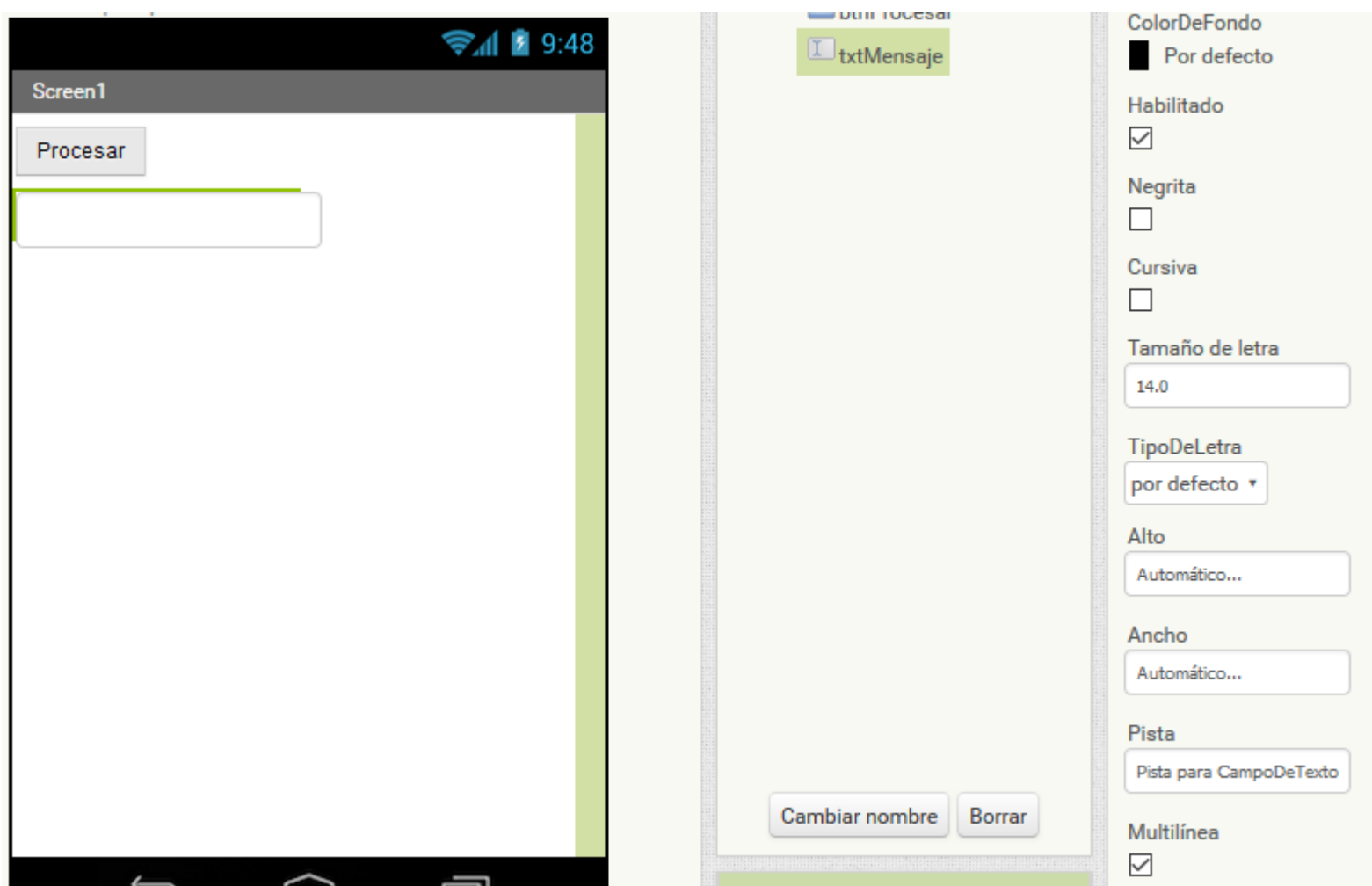


Imagen 104: Campo texto tiene activado el multilínea

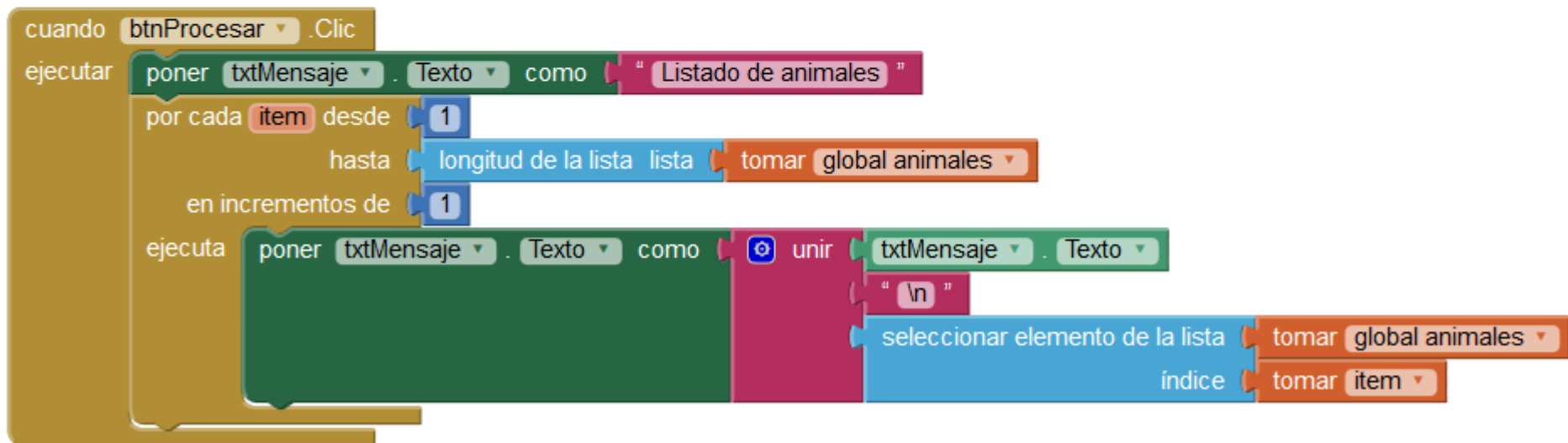
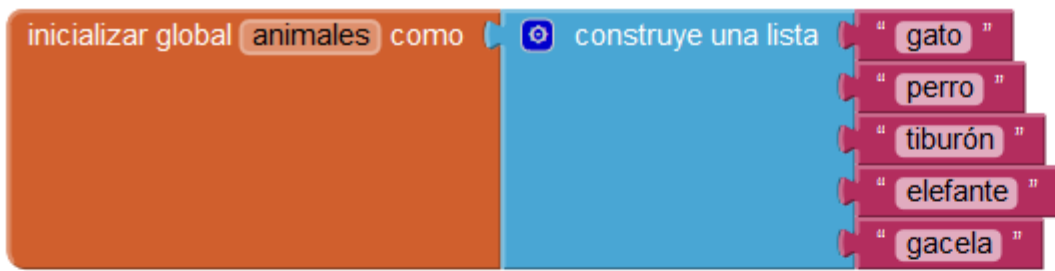


Imagen 105: Volcar una lista



Imagen 106: Muestra el listado

D. Reemplazar un ítem de una lista

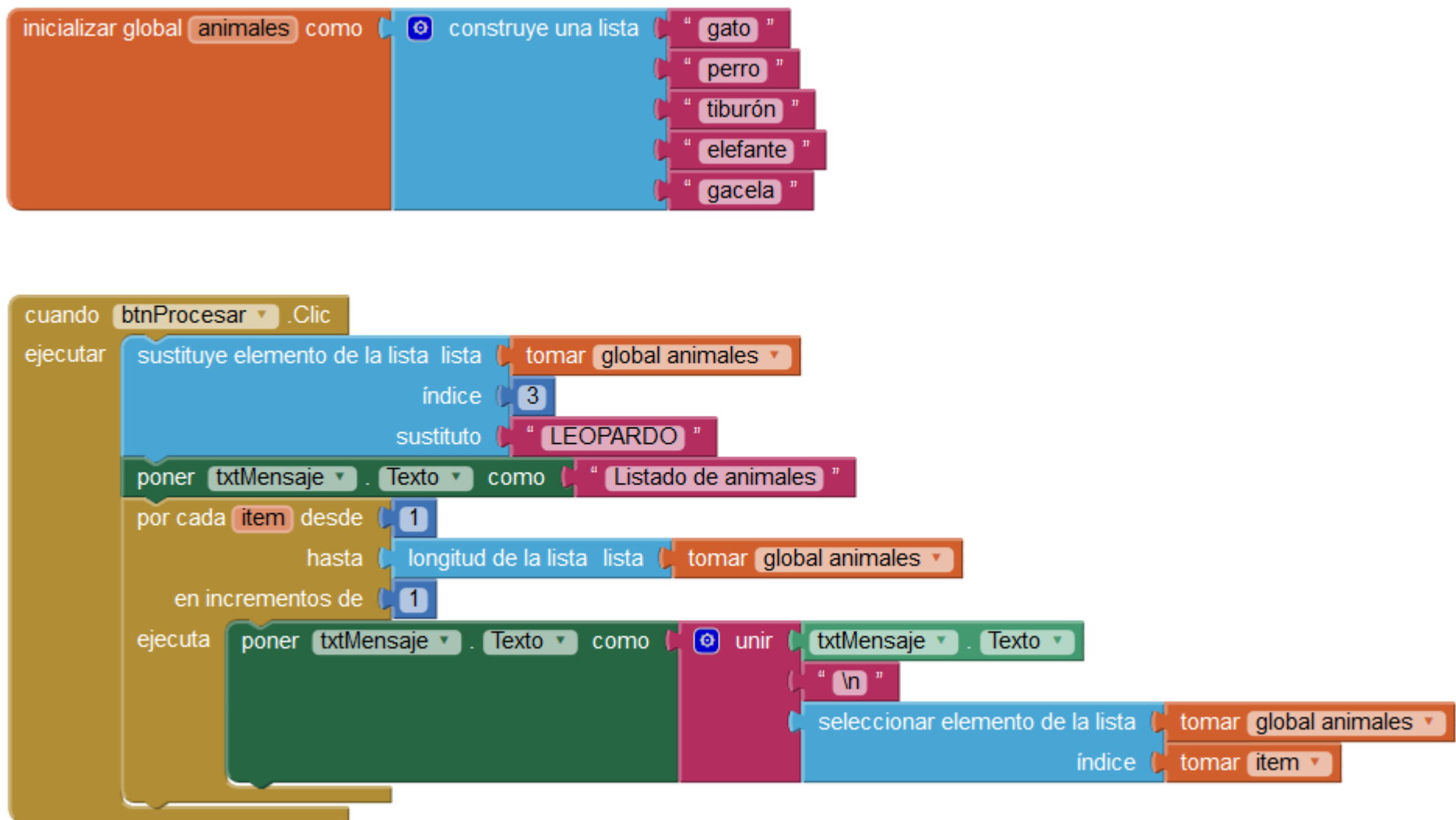


Imagen 107: Reemplaza el tercer ítem de la lista definida

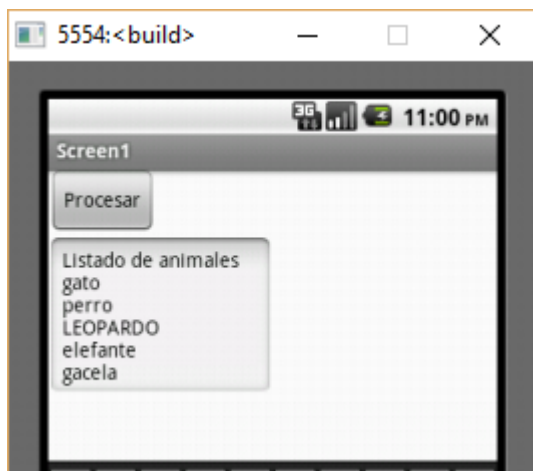


Imagen 108: Reemplazó el tercer ítem

E. Verifica si hay un ítem en una lista

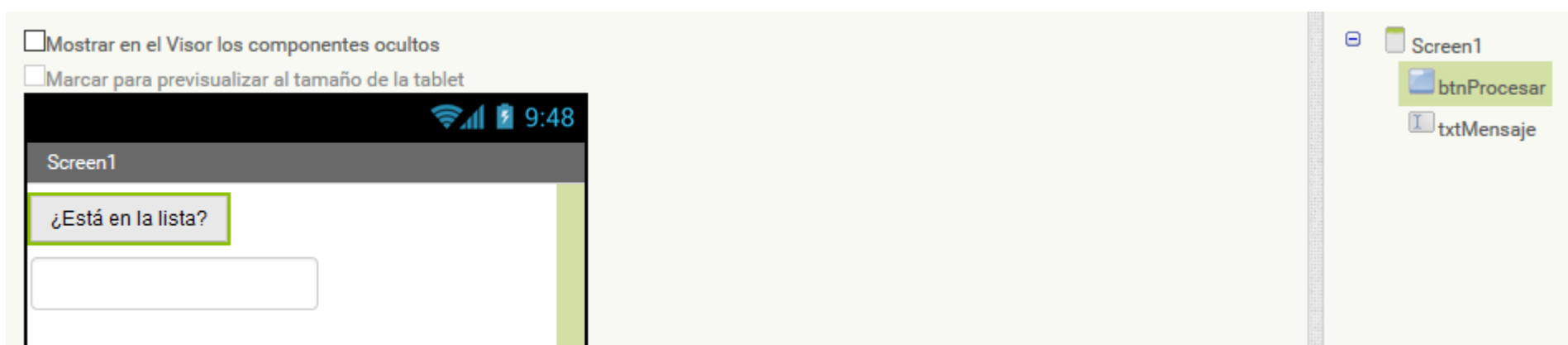


Imagen 109: Diseño



Imagen 110: Chequea si el ítem "elefante" está en la lista retornando "true" o "false". Se considera mayúscula y minúscula

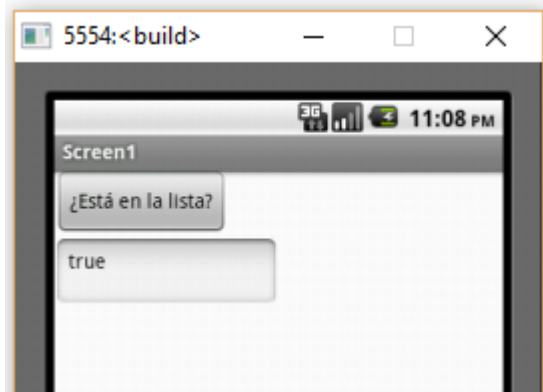


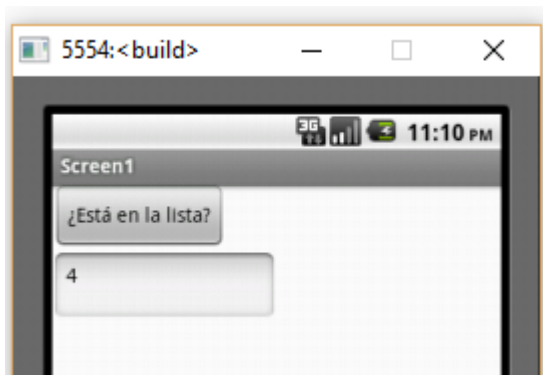
Imagen 111: Ejecución

F. Retorna en qué posición se encuentra un determinado ítem en la lista

Si el ítem no se encuentra en la lista retorna cero (0).



Imagen 112: Retorna que "elefante" está en la posición 4.



G. Retorna si la lista está llena o vacía

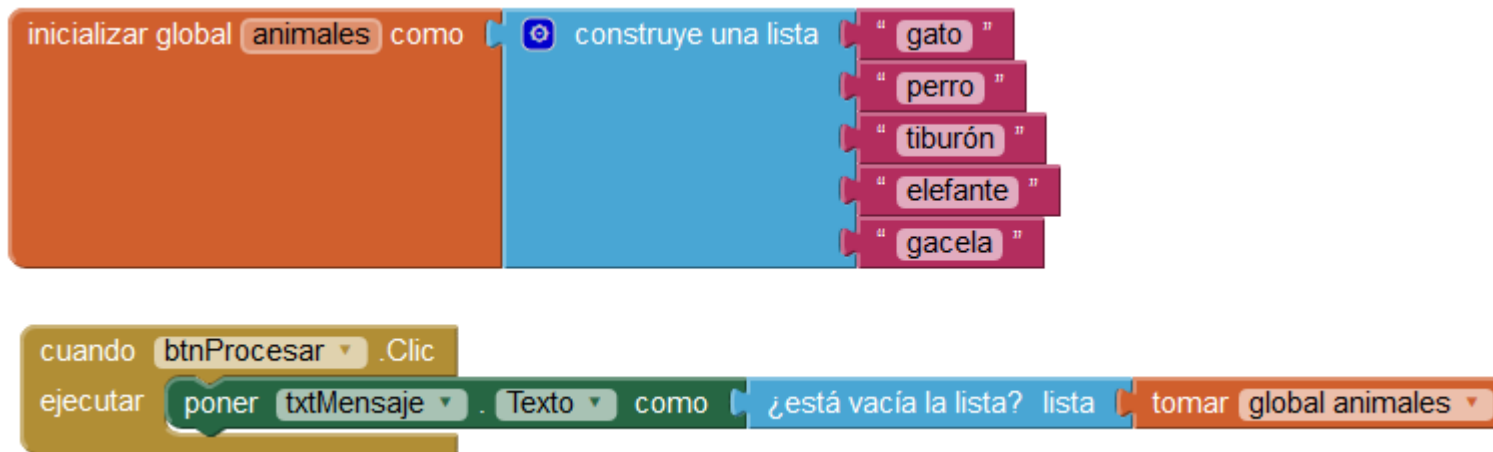


Imagen 113: Retorna "false" porque la lista tiene ítems.

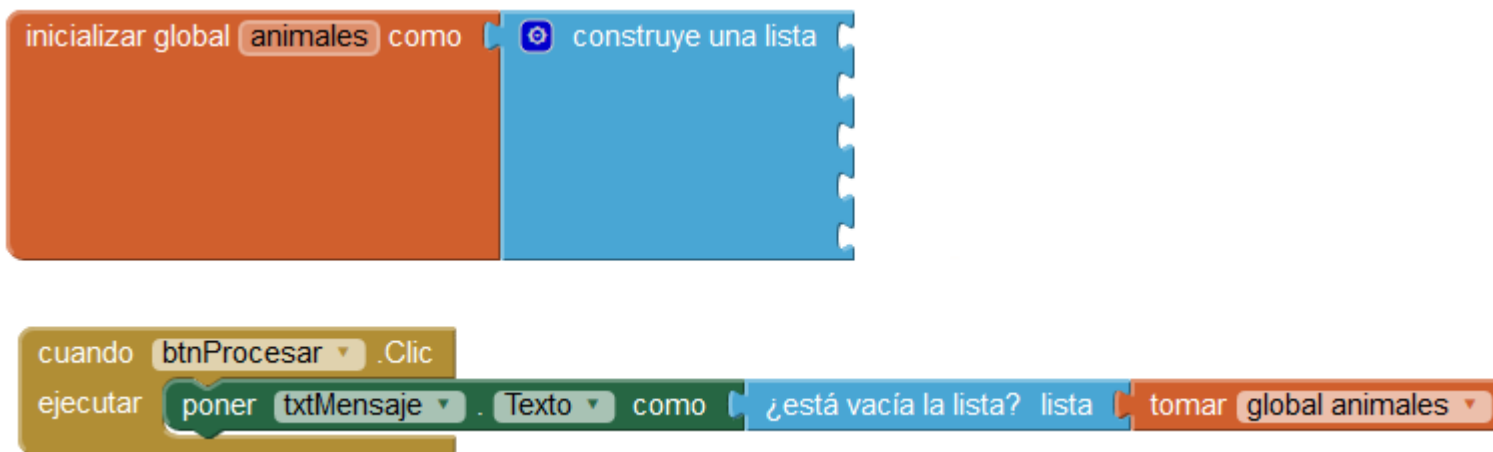
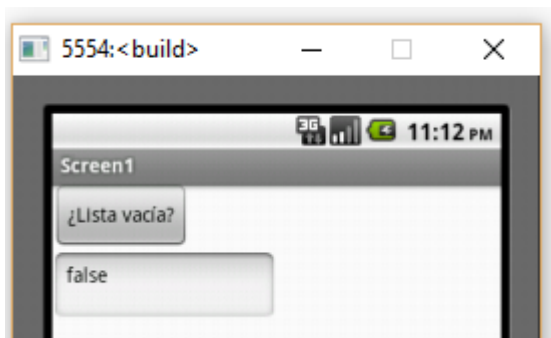
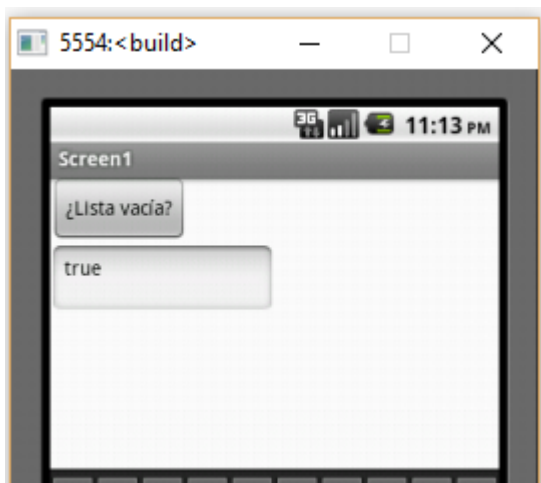


Imagen 114: Retorna "true" porque la lista está vacía.



H. Inserta un ítem en determinada posición de la lista

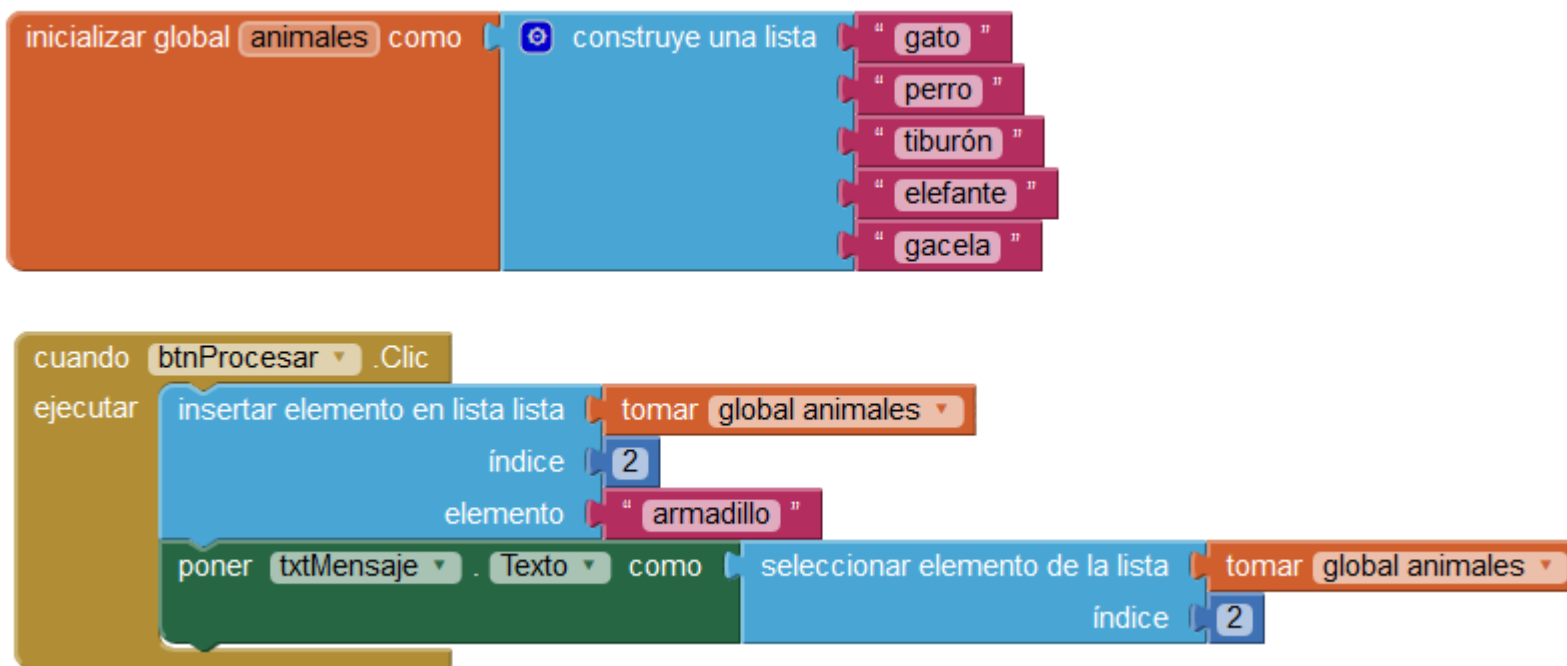
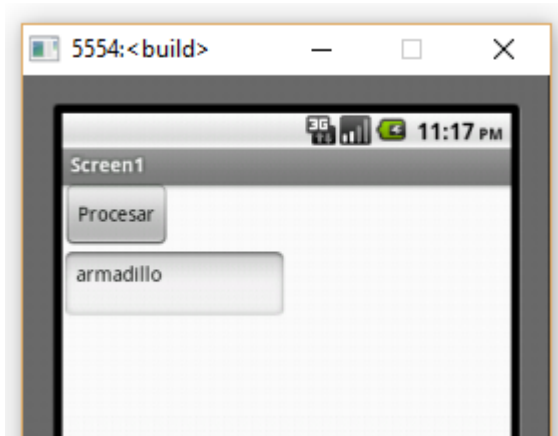


Imagen 115: En la posición 2 se inserta "armadillo" corriendo a "perro" a la tercera posición



I. Quita un elemento de la lista

En este ejemplo se hace uso de variables locales. Obsérvese como es su tratamiento.

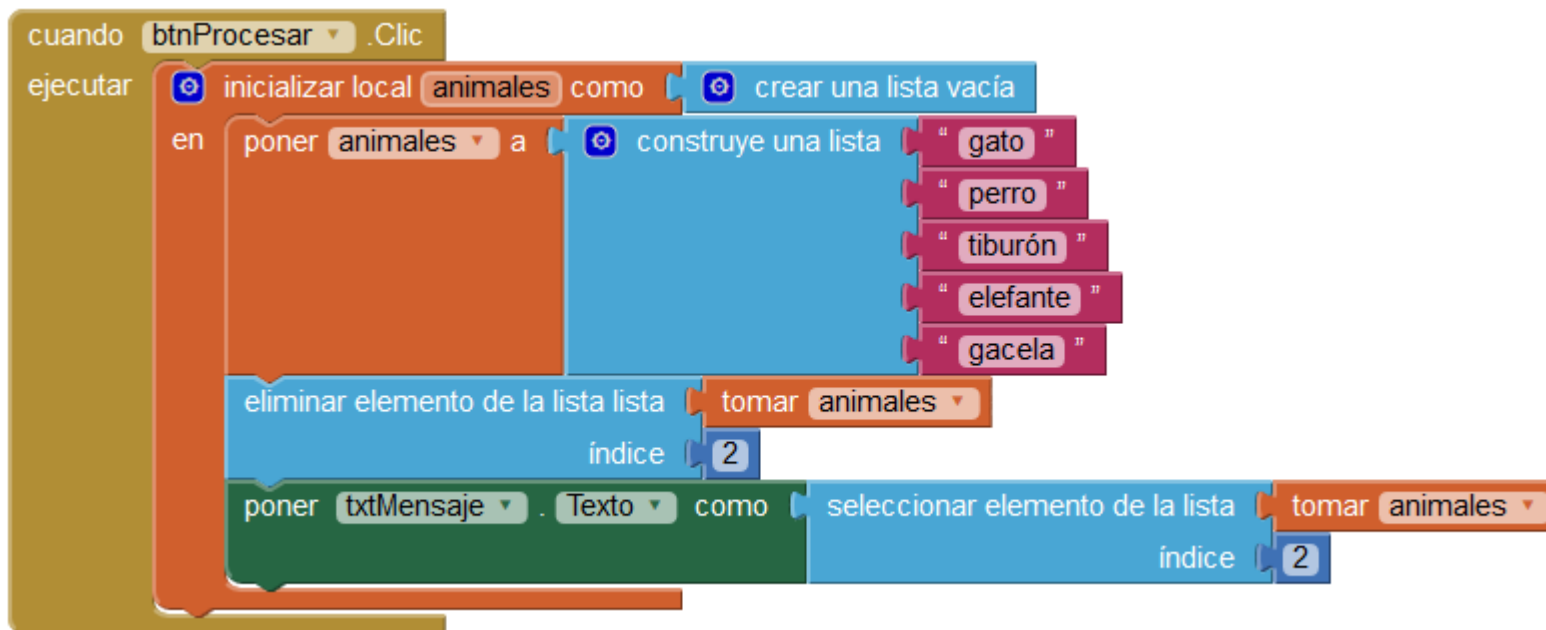
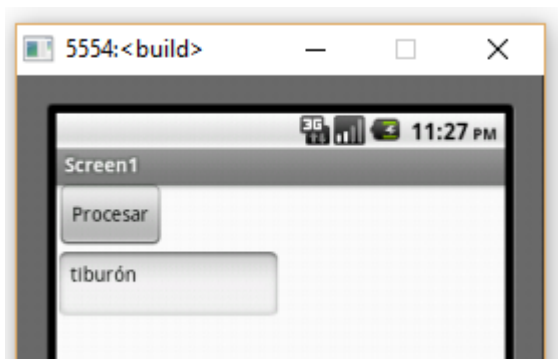


Imagen 116: Quita el segundo elemento de la lista, luego el tercero ocupa su lugar



J. Copia una lista en otra

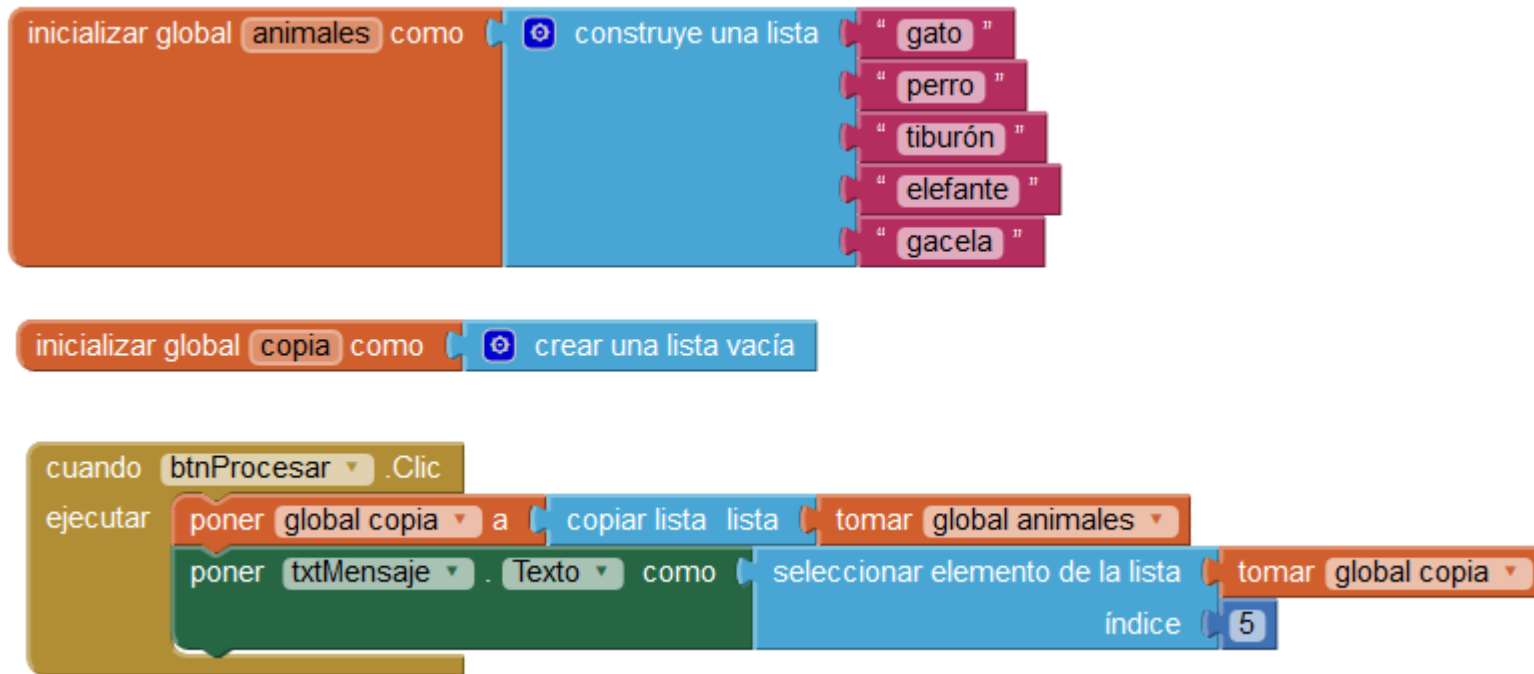
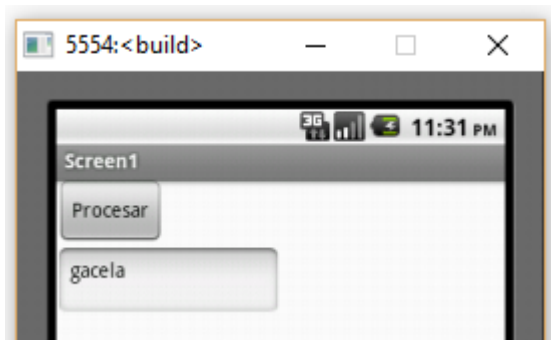


Imagen 117: Copia una lista en otra



K. Adiciona una lista a otra lista

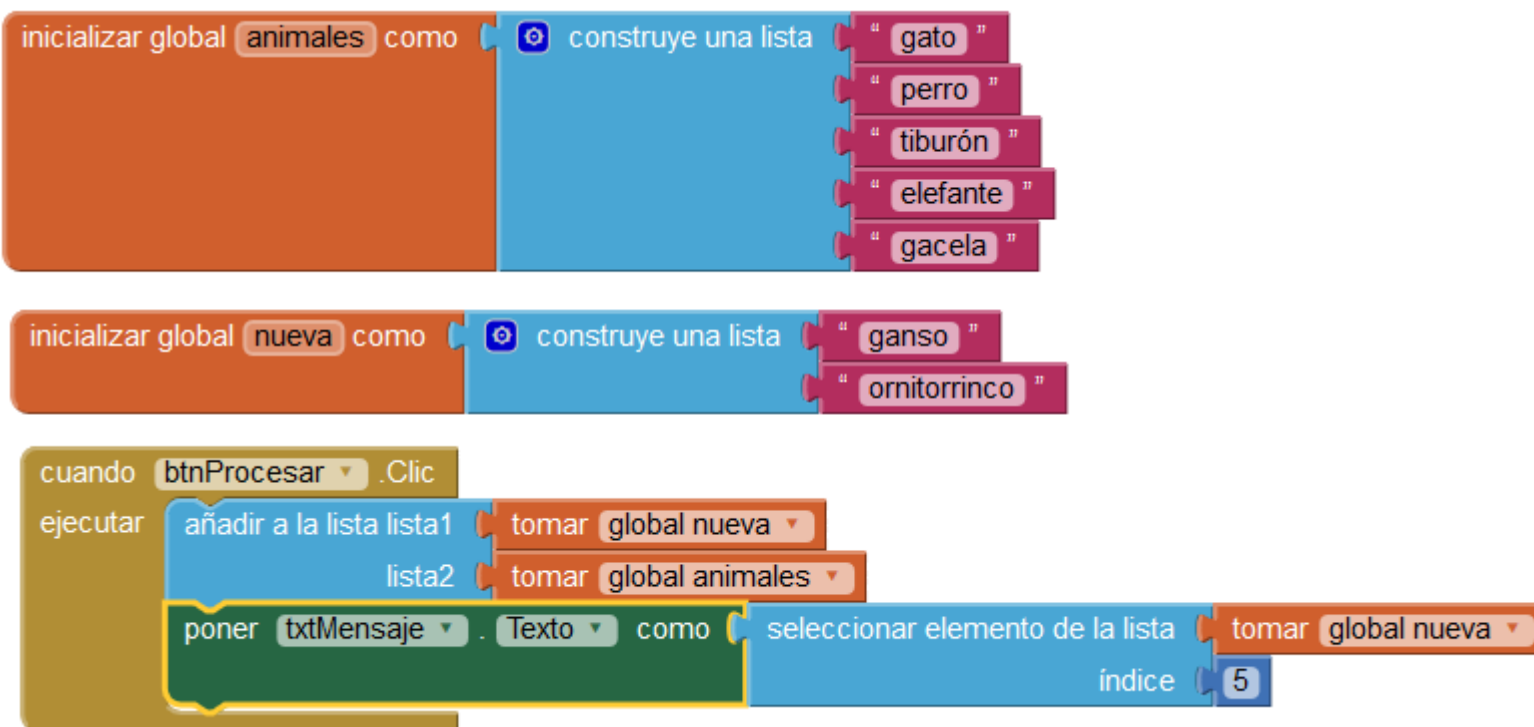
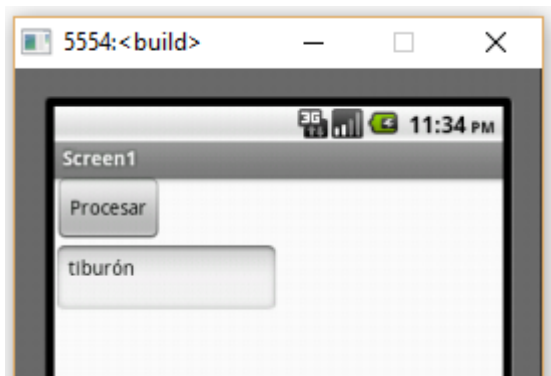


Imagen 118: La lista "nueva" recibe los ítems de la lista "animales"



L. ¿Es determinado objeto una lista?

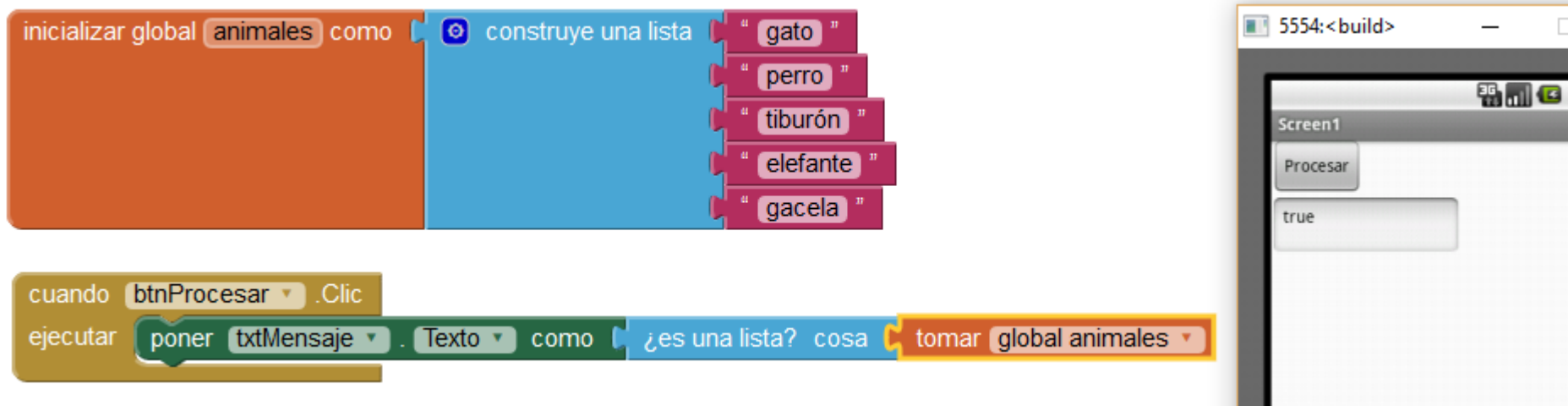


Imagen 119: ¿Es determinado objeto una lista? "true" si es una lista, "false" si no.

Paso 13. Procedimientos y Funciones

MIT App inventor permite hacer uso de procedimientos y funciones

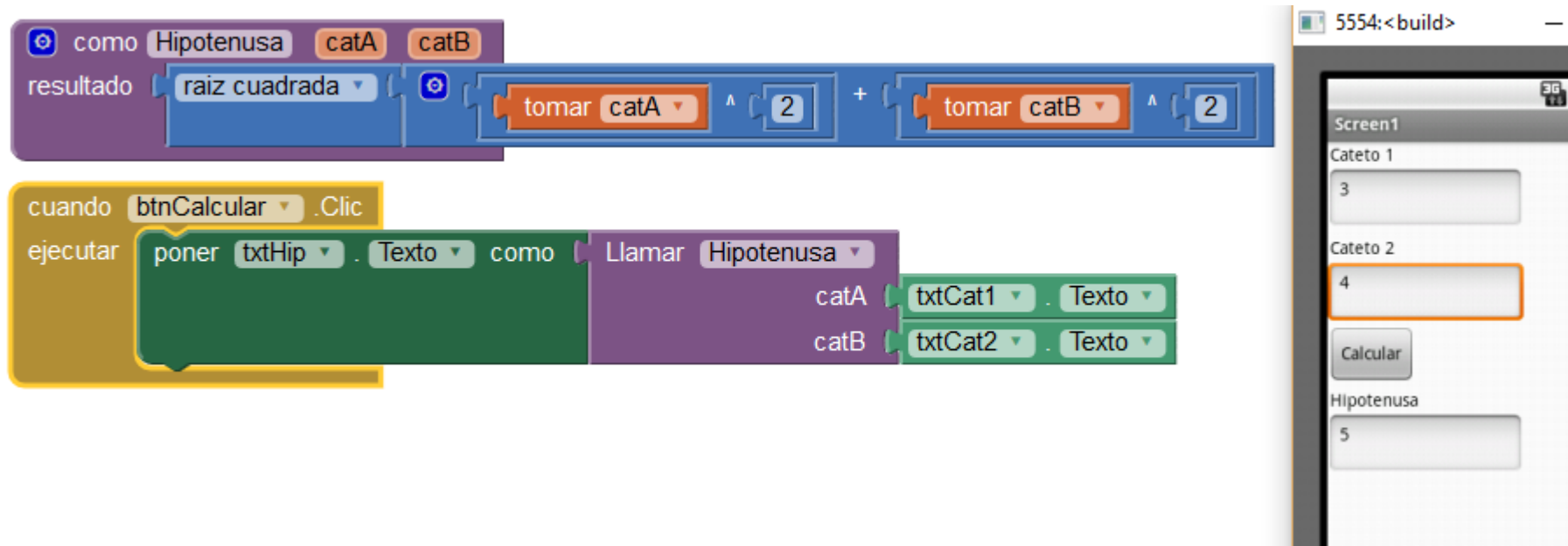


Imagen 120: Uso de funciones

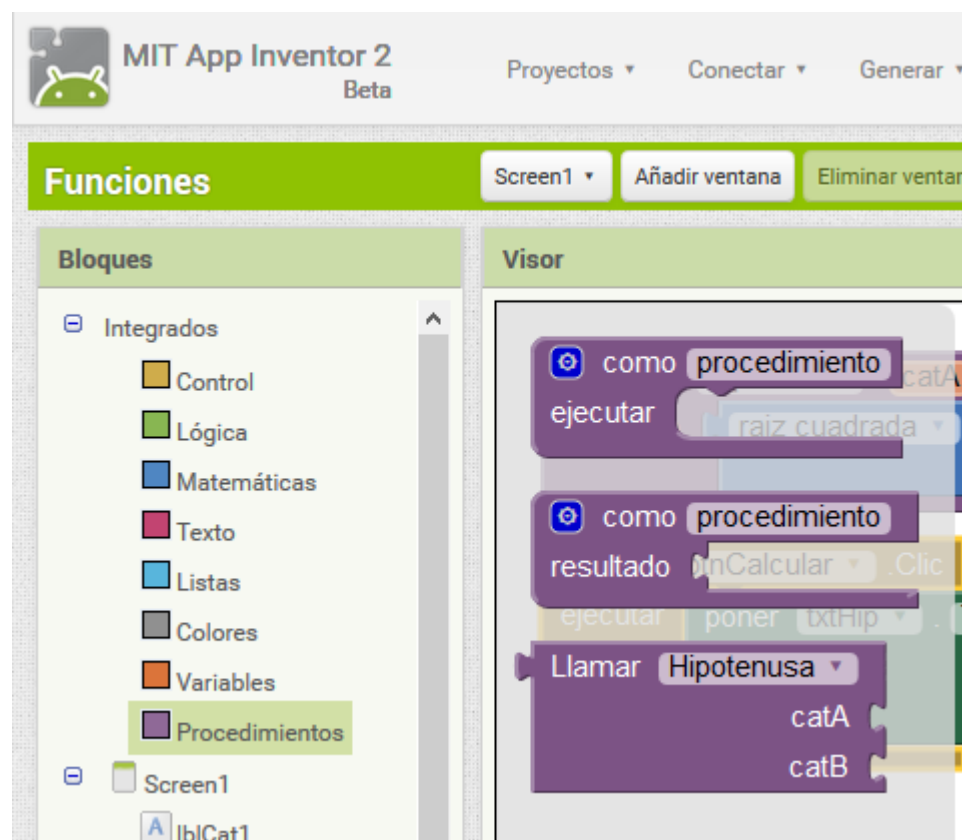


Imagen 121: Los bloques están en procedimientos

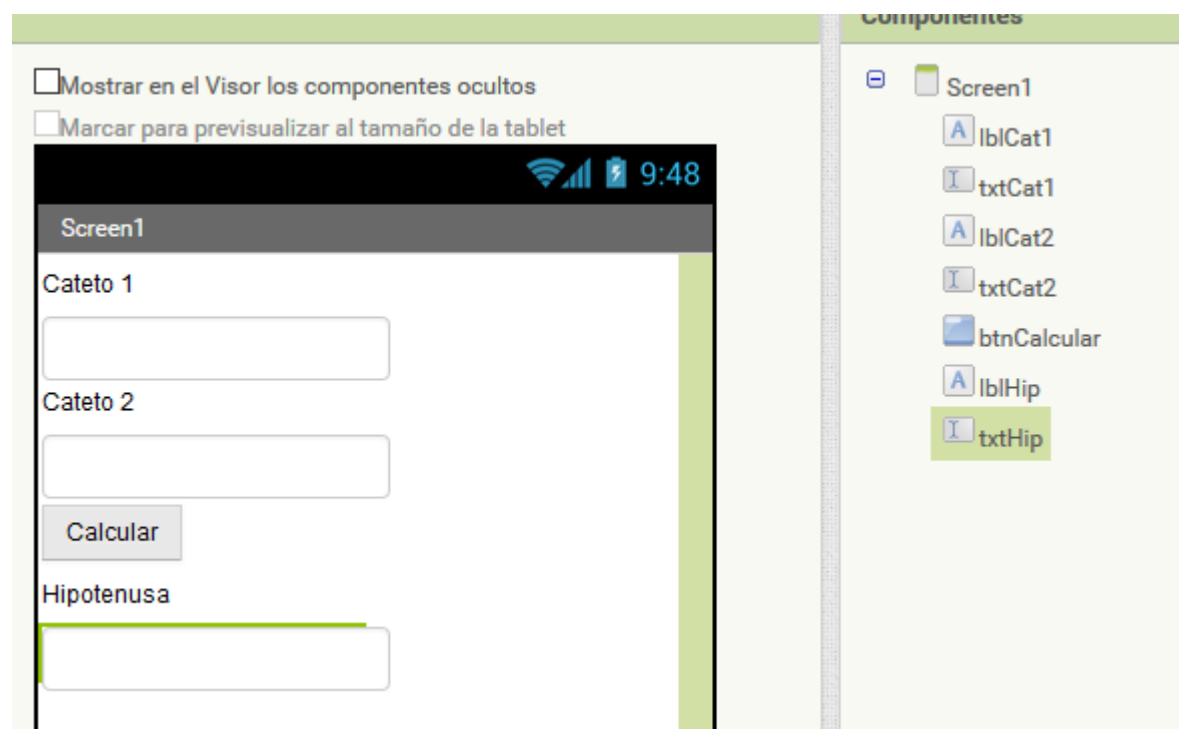


Imagen 122: El diseño

Paso 14. Gráficos

Para hacer gráficos (líneas, puntos, círculos) se requiere un objeto llamado Lienzo, el cual se arrastra a la ventana de diseño

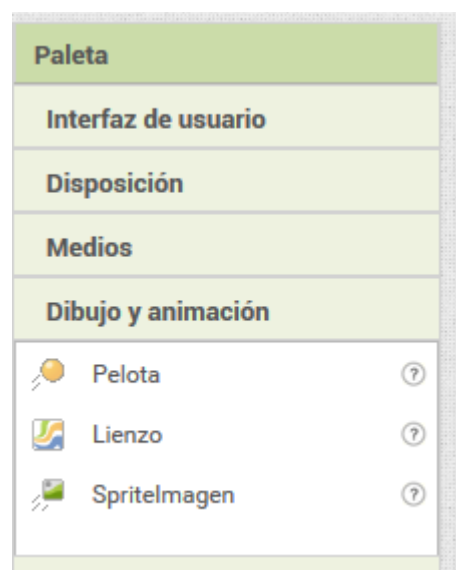


Imagen 123: Se requiere un objeto llamado "Lienzo" ubicado en "Dibujo y animación"

Se recomienda ajustar las propiedades del lienzo en Alto y en Ancho a "Ajustar al contenedor..."

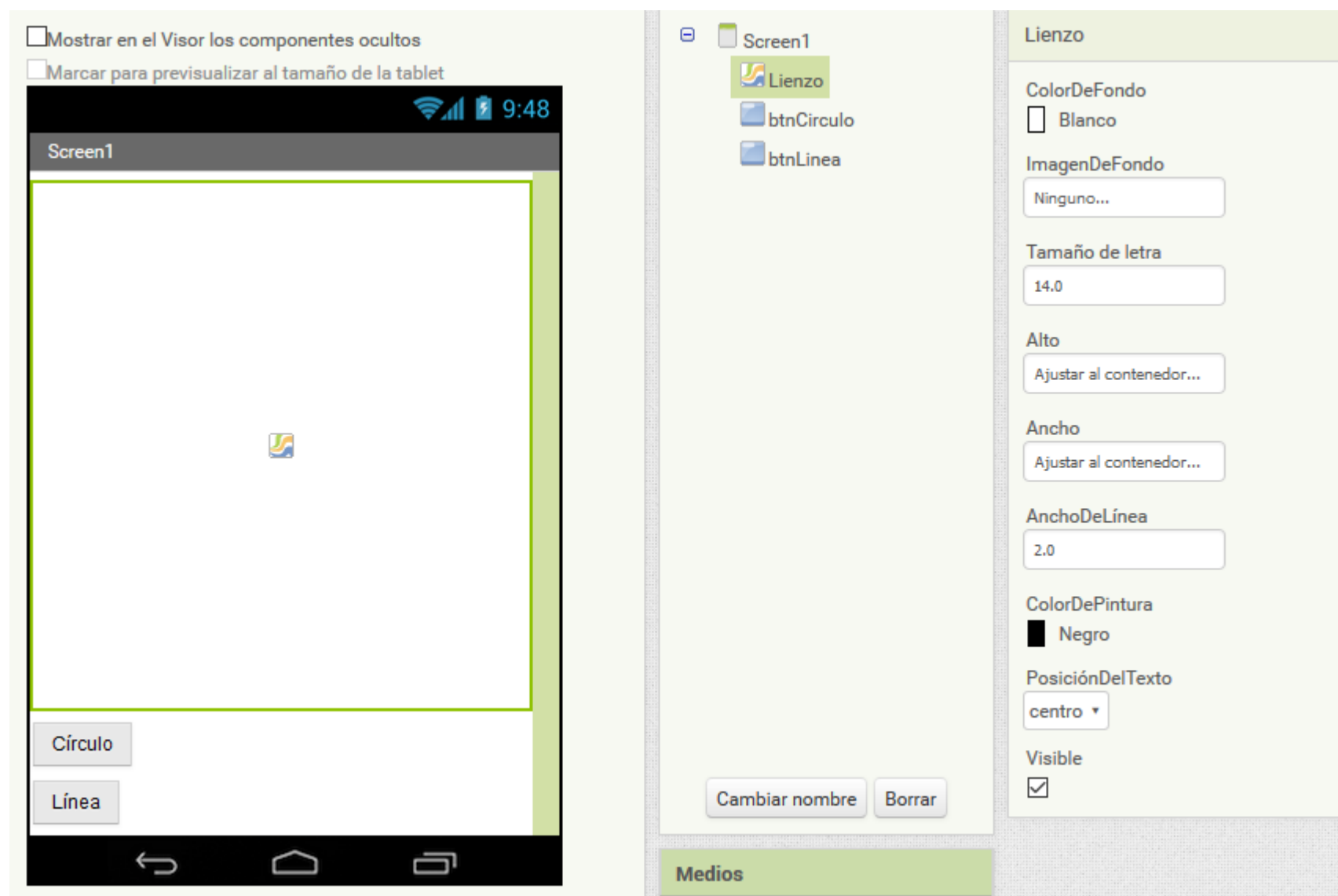


Imagen 124: Objeto "Lienzo" junto con dos botones

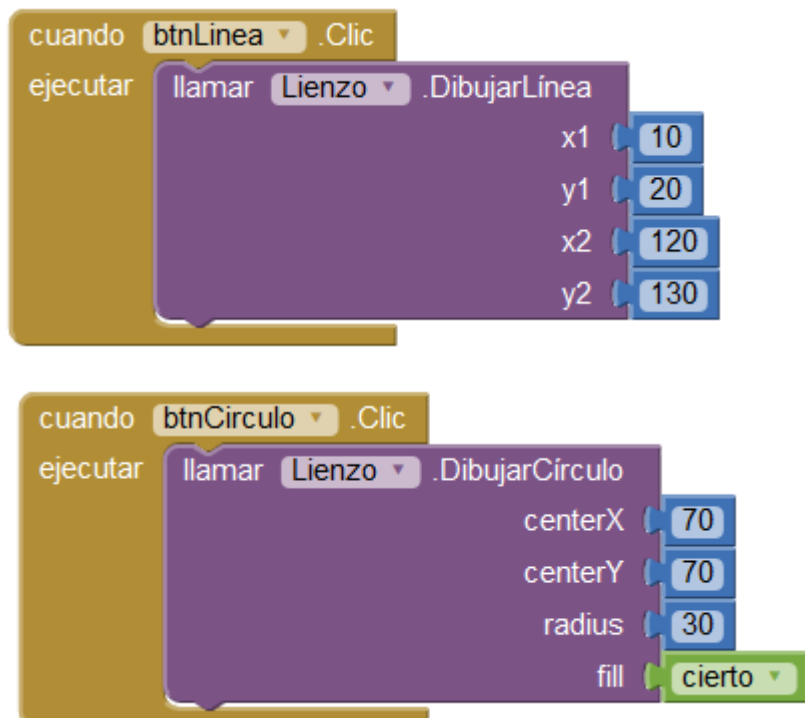


Imagen 125: Cuando se presionan los botones, desde el objeto "Lienzo", se hace llamado a las funciones de dibujado

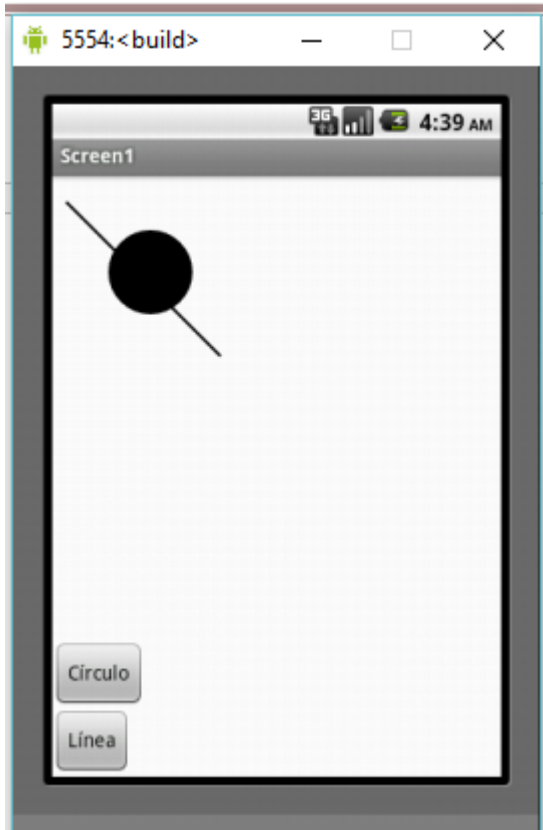


Imagen 126: Así se dibuja en el objeto "Lienzo"

A. Dibujar puntos, líneas y círculos

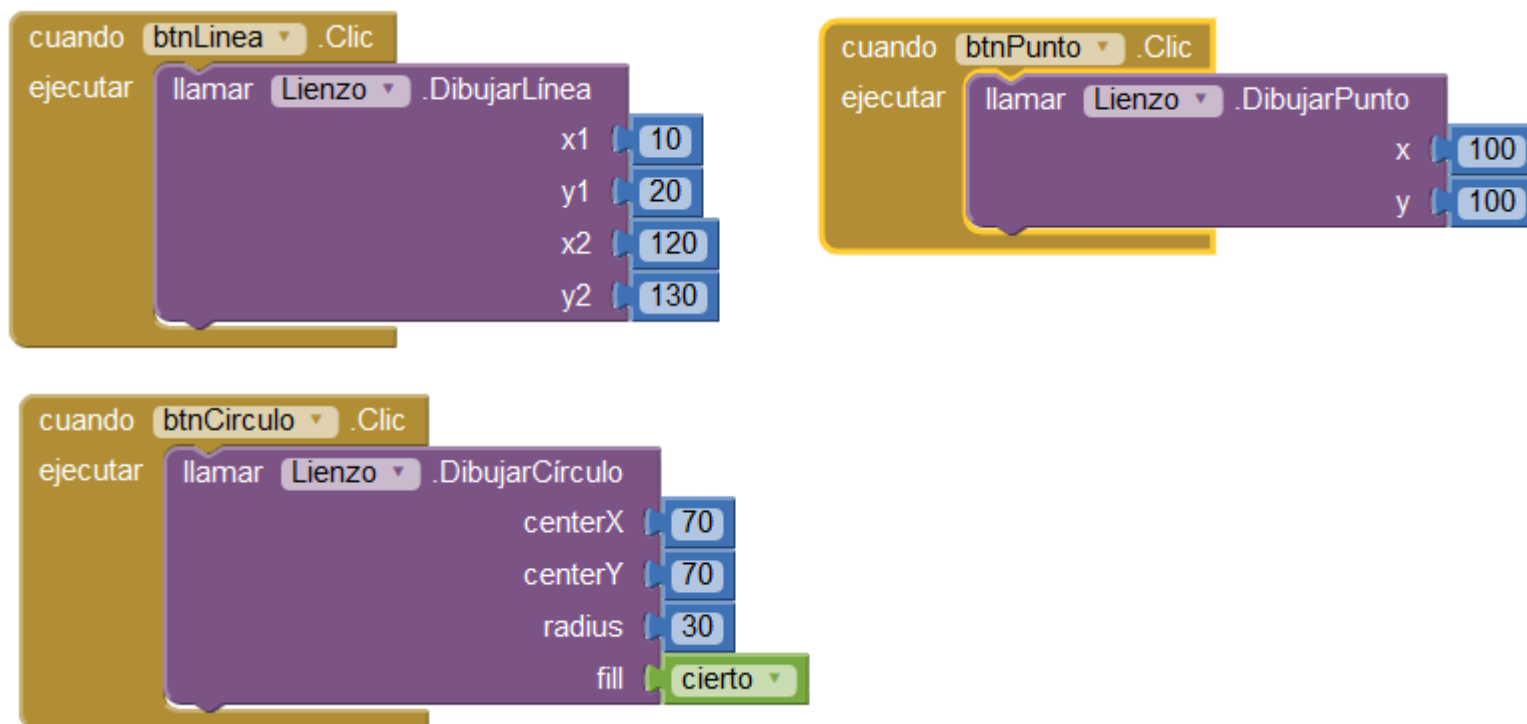


Imagen 127: La función "DibujarPunto" permite dibujar puntos en determinada posición

B. Dibujar textos

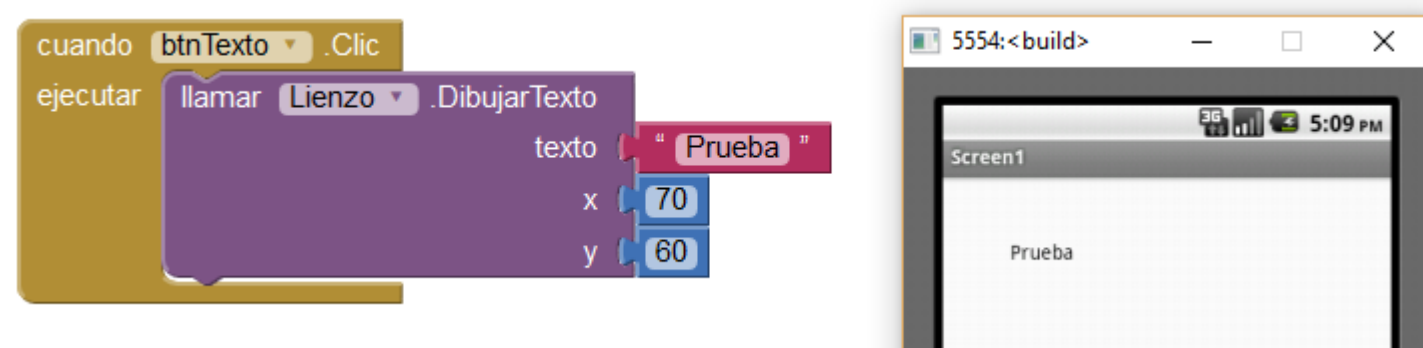


Imagen 128: La función "DibujarTexto" permite dibujar textos en determinada posición

C. Dibujar texto girado en un determinado ángulo

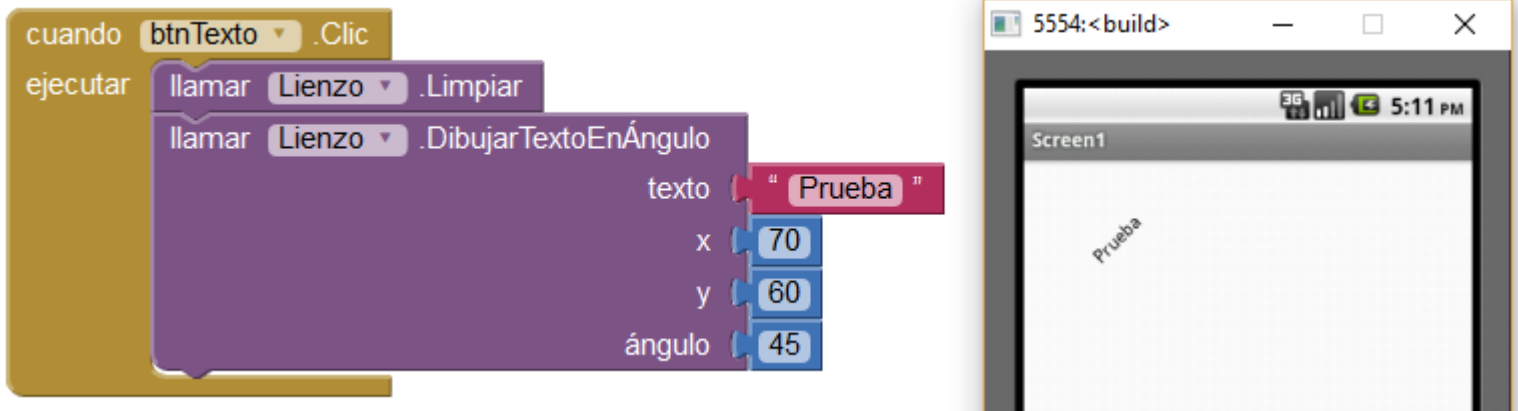


Imagen 129: Con la instrucción "Limpiar" borra el lienzo, con la instrucción "DibujarTextoEnÁngulo" dibuja el texto en un ángulo

D. Tamaño y color del texto



Imagen 130: Con "Tamaño de letra" cambia el tamaño de la fuente y con "Color de Pintura" cambia el color de lo que se dibuja

E. Fondo del lienzo en un color determinado

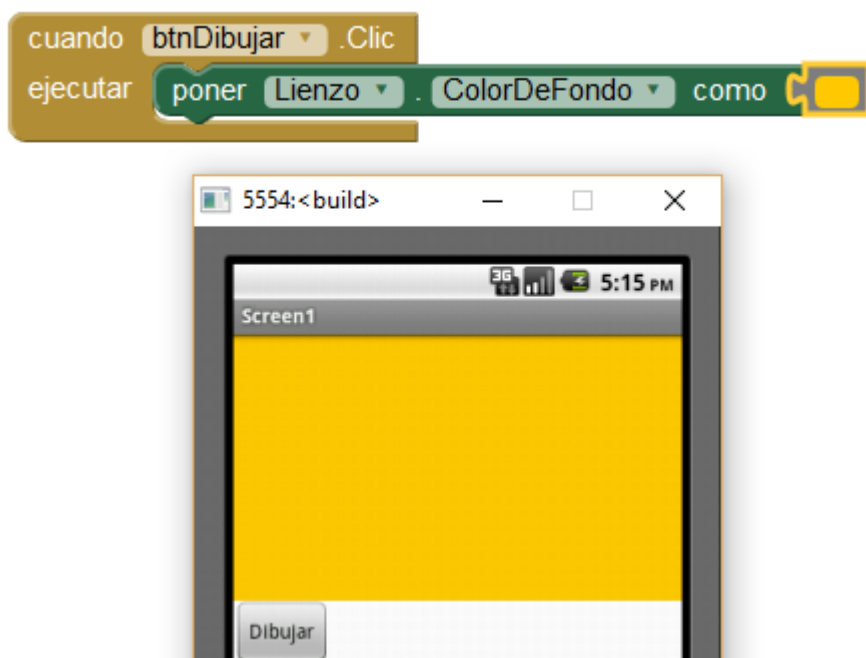


Imagen 131: Con "ColorDeFondo" se puede cambiar el color de fondo del lienzo

F. Dibujando una línea con determinado ancho

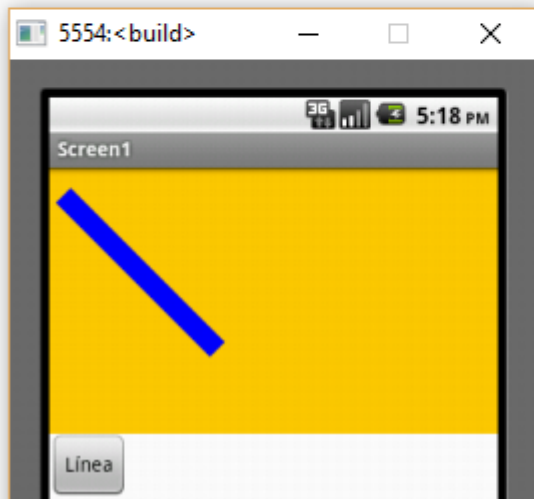
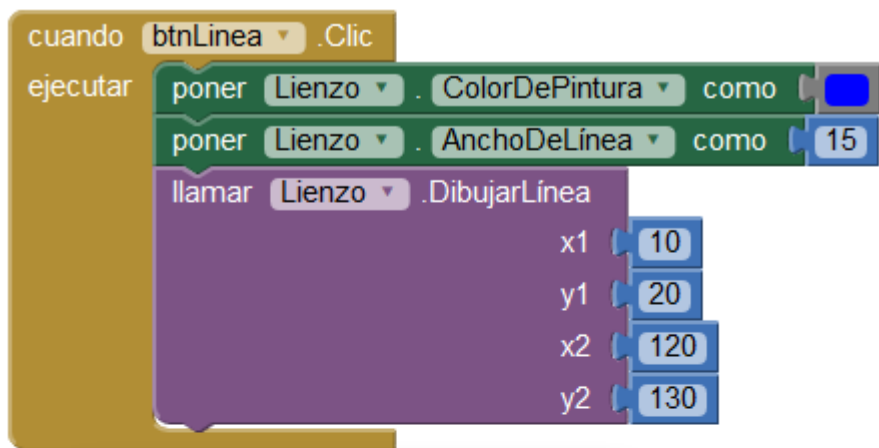


Imagen 132: Dibuja de color azul la línea con determinada anchura "AnchoDeLínea"

G. Imagen de fondo

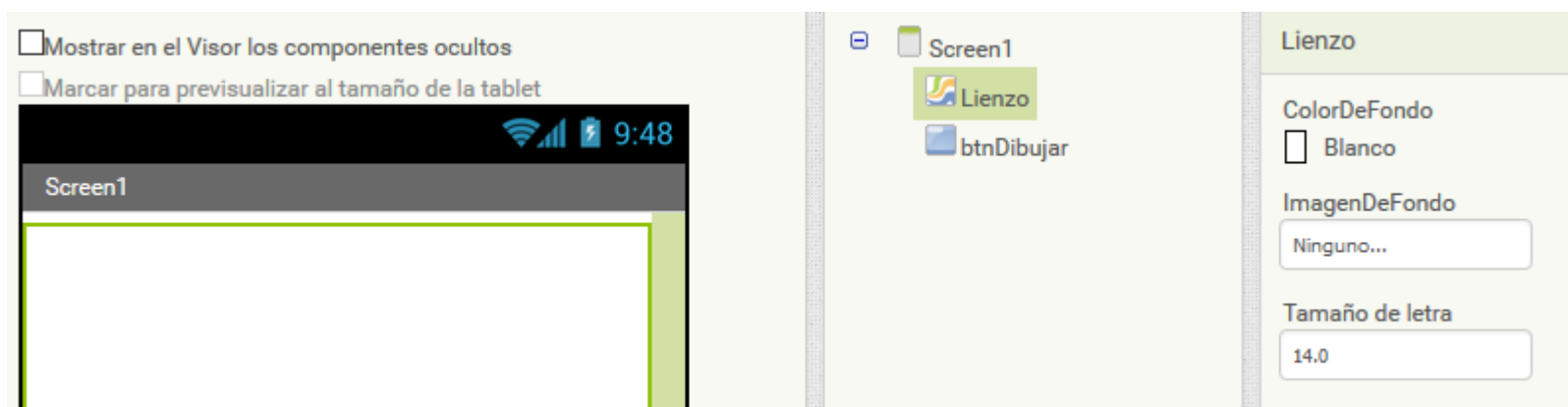


Imagen 133: Puede poner una imagen de fondo al lienzo, de clic en ImagenDeFondo

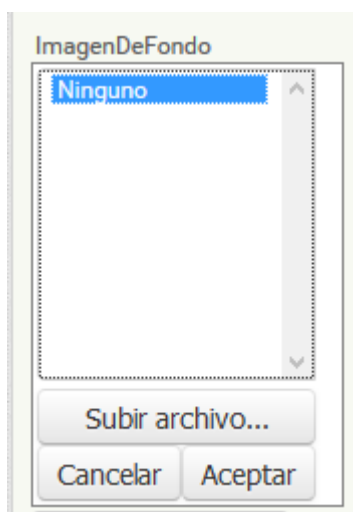


Imagen 134: De clic en "Subir archivo..."

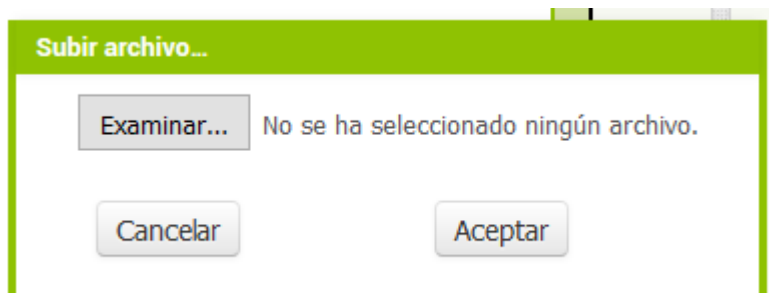


Imagen 135: De clic en "Seleccionar archivo"

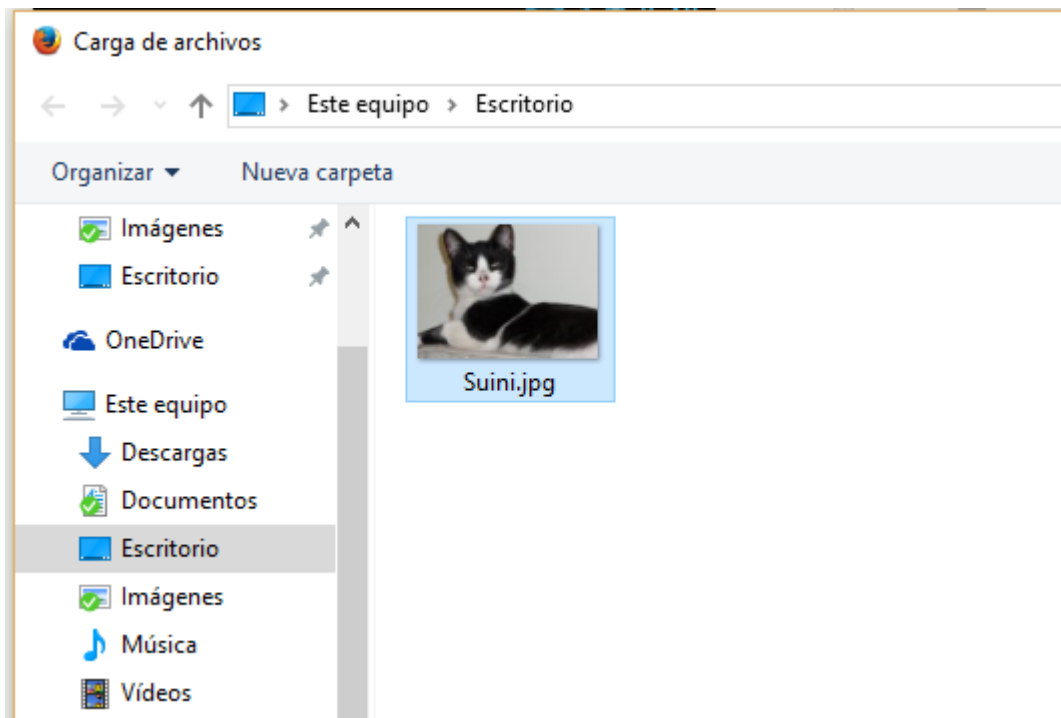


Imagen 136: Seleccione una imagen. Recomendado: De tamaño pequeño

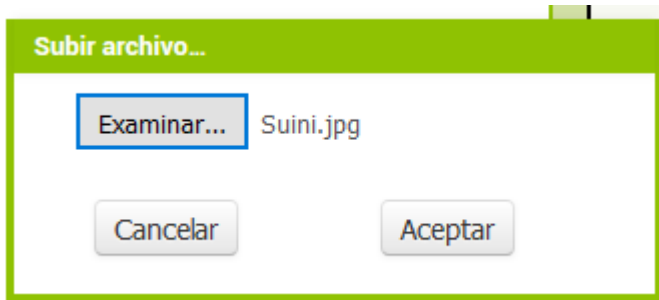


Imagen 137: Imagen cargada. De clic en "OK"

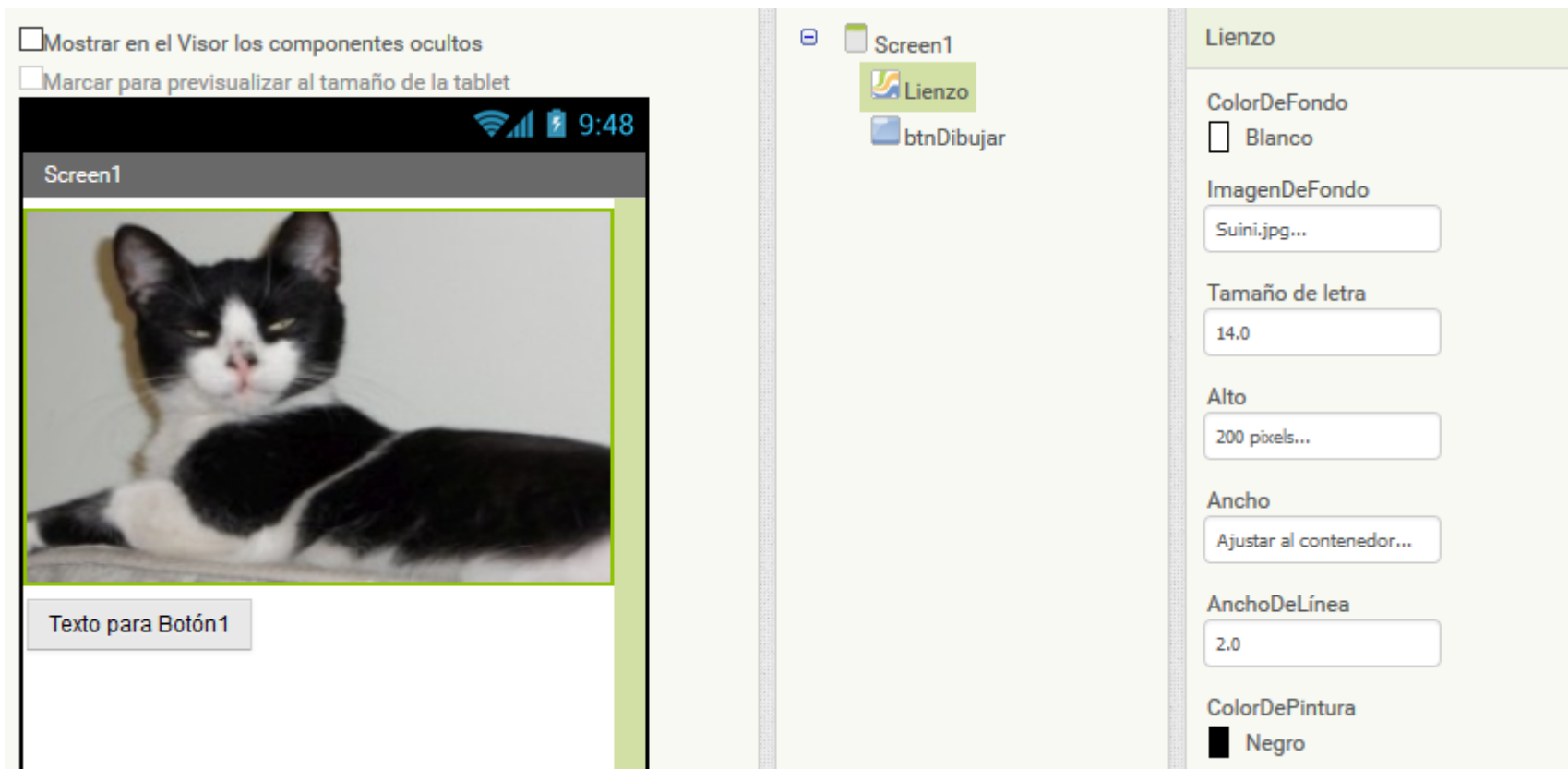


Imagen 138: Ya hay una imagen de fondo



Imagen 139: Ejecución con imagen de fondo

H. Dibujar círculos con clic de ratón (inicio de captura de eventos)

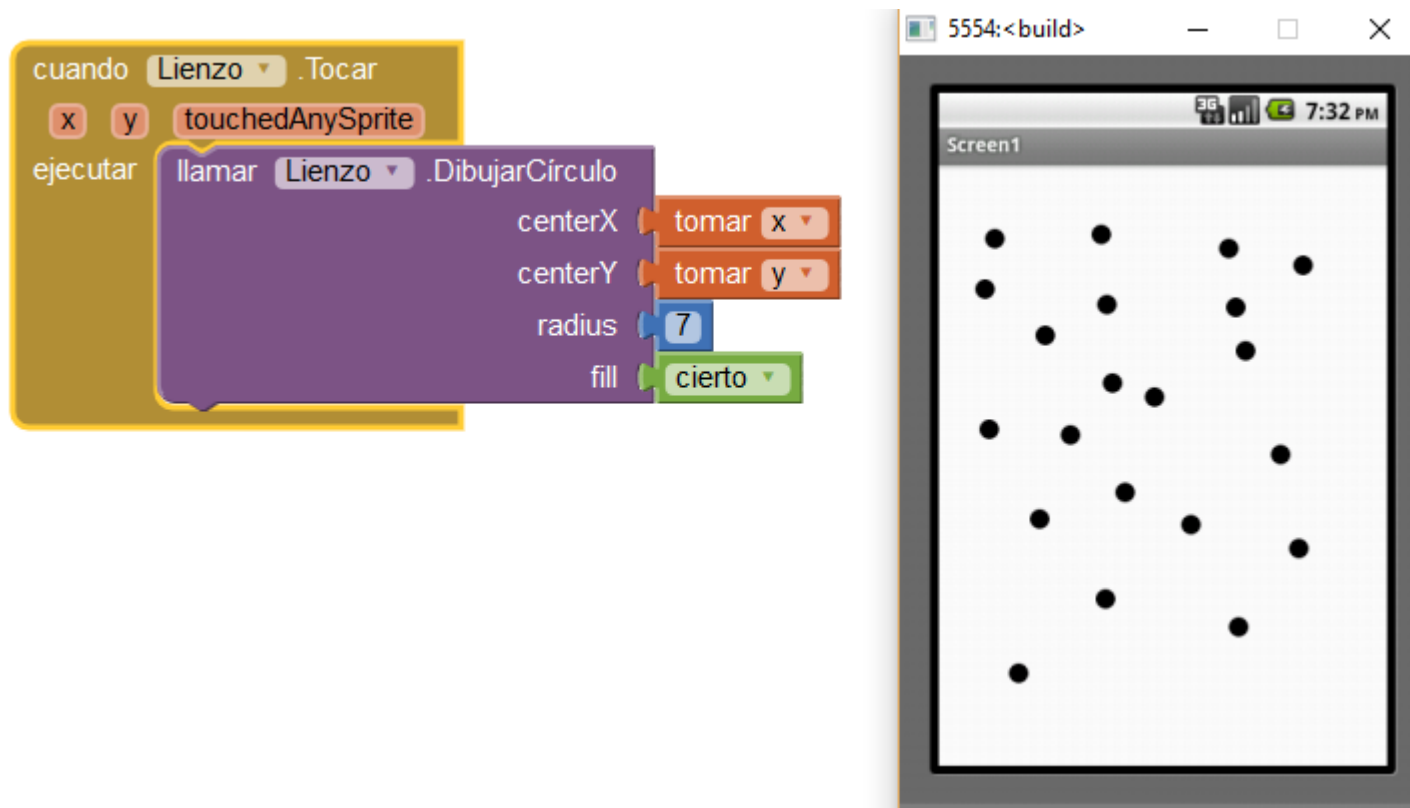


Imagen 140: Captura de eventos

El evento "Tocar" define las variables "x" y "y" que son las coordenadas del plano donde el usuario da clic. Luego la sentencia que se ejecuta al hacer "Tocar", que en el ejemplo es "DibujarCírculo" puede hacer uso de esas variables. El usuario al hacer clic en algún punto del lienzo, va dibujando círculos rellenos.

I. Hacer un sencillo programa de dibujo (captura de eventos)

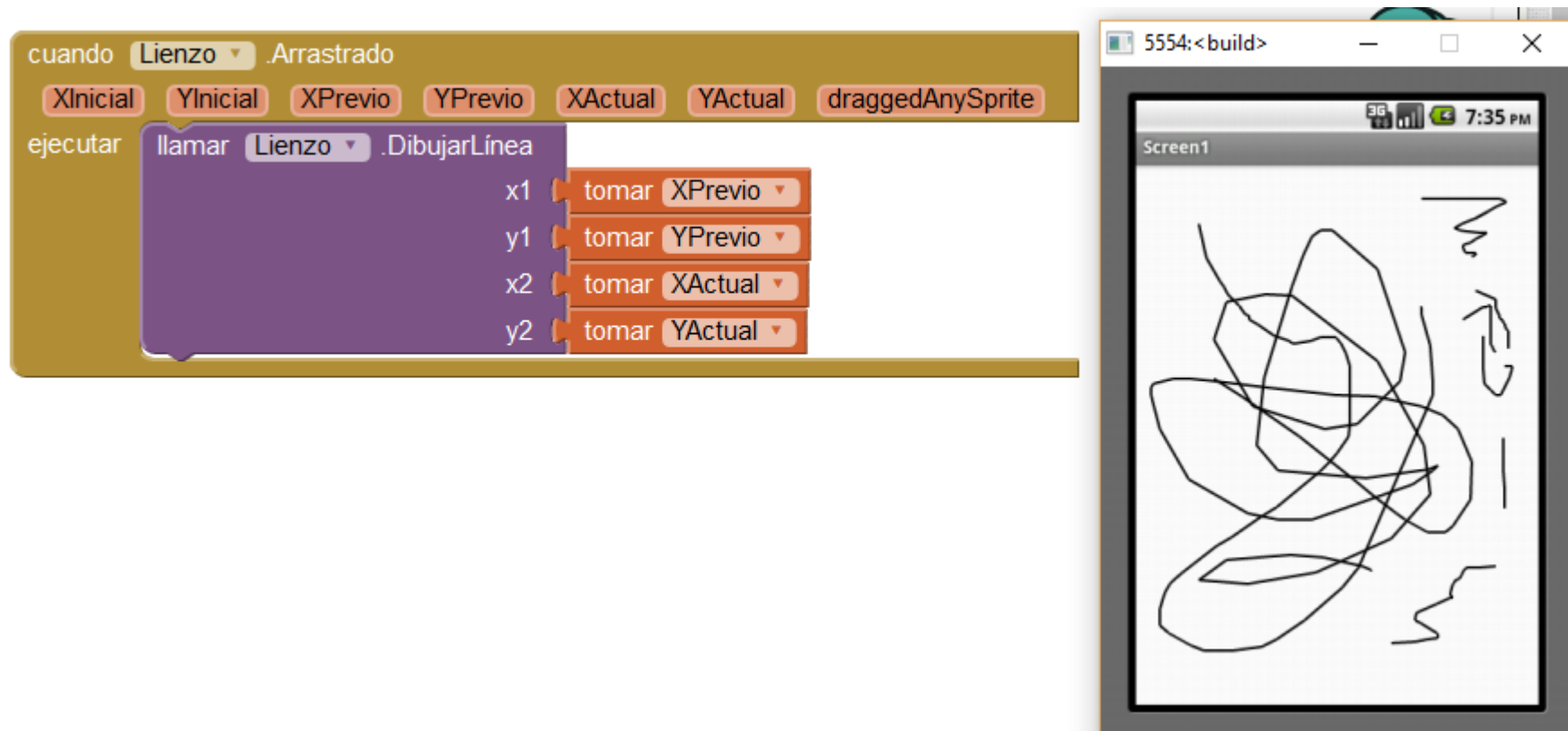


Imagen 141: Al mantener presionado el botón izquierdo del ratón, se dibuja lo que quiere.

Paso 15. Animación

Hay varias formas de hacer animación, se muestra usando un componente de tiempo y otro usando las propiedades de un objeto gráfico

A. Usando un componente de tiempo

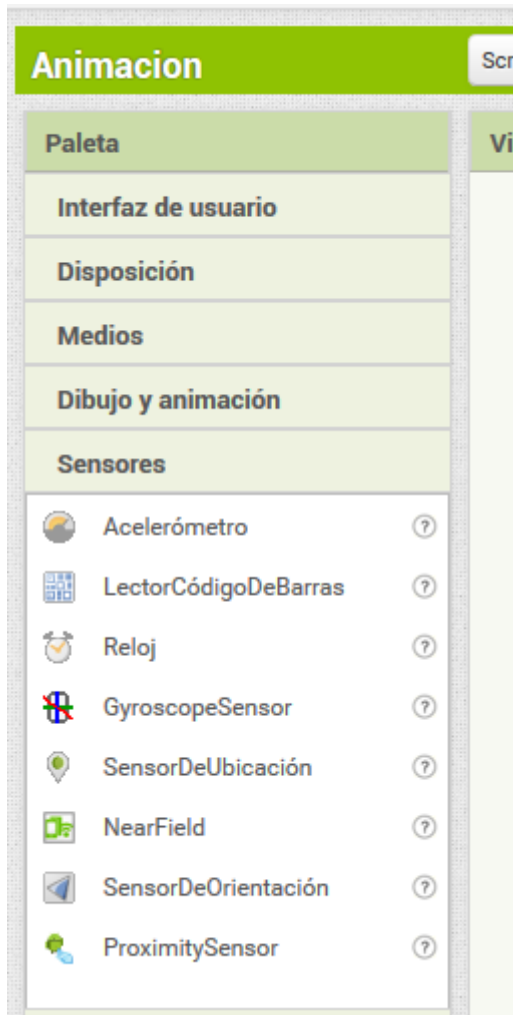


Imagen 142: Objeto "Reloj" para manejar la animación

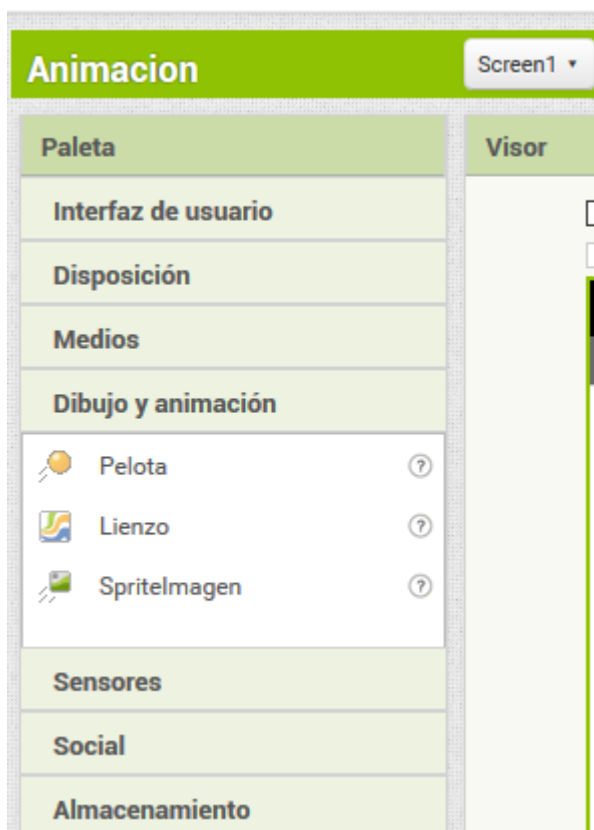


Imagen 143: Se utiliza el objeto "Ball" que será lo que se va a animar

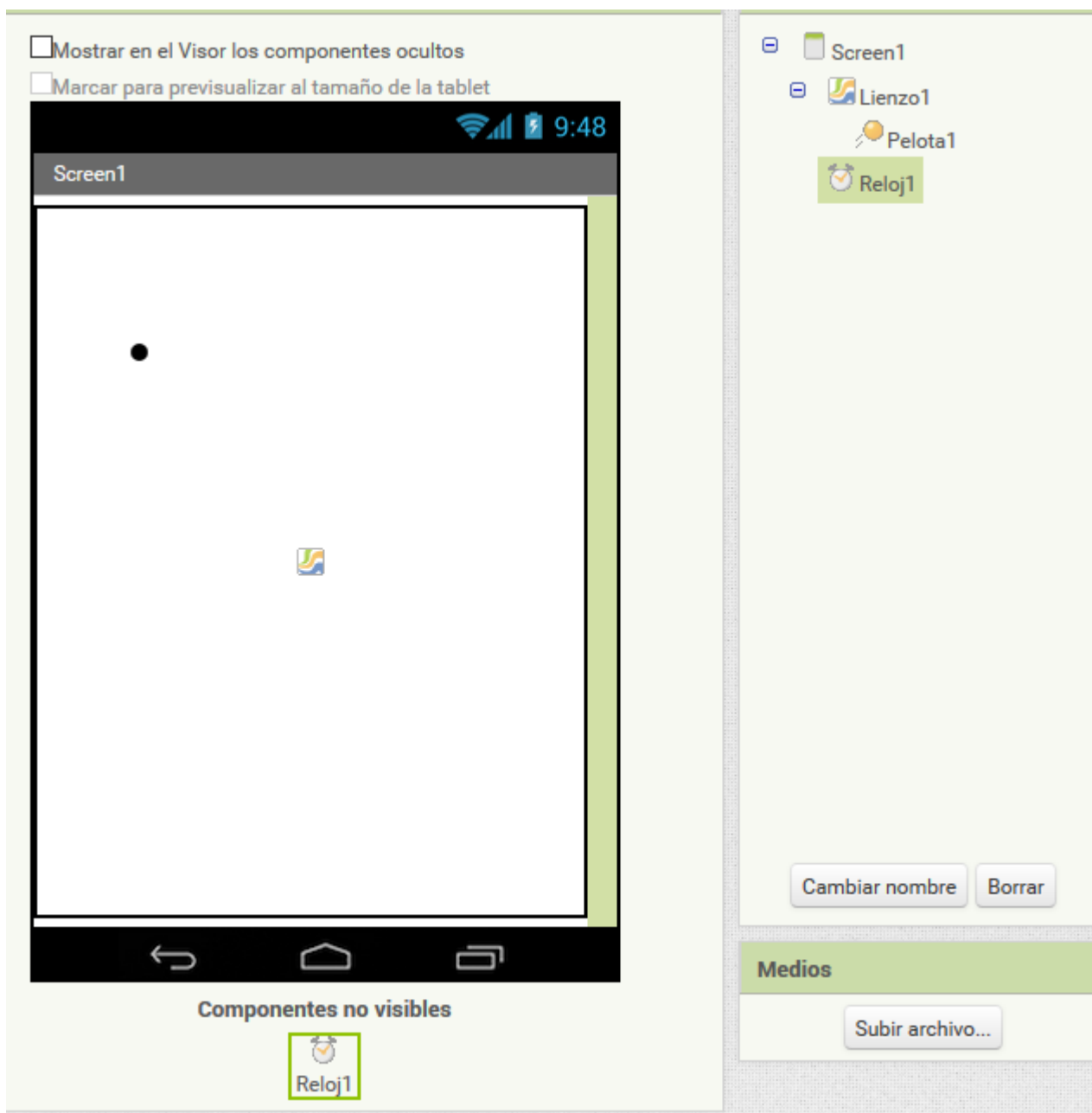


Imagen 144: El proyecto es un objeto "Lienzo", un objeto "Reloj" y un objeto "Pelota"

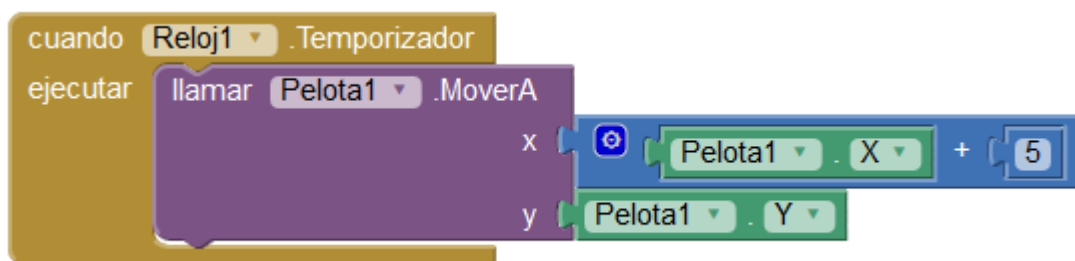


Imagen 145: El algoritmo de animación. Mover el objeto "Pelota" horizontalmente.

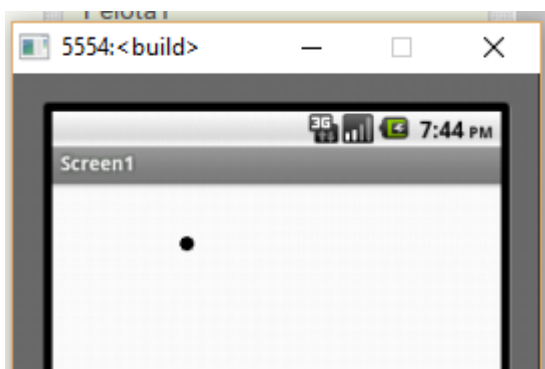


Imagen 146: Un "frame" de la animación

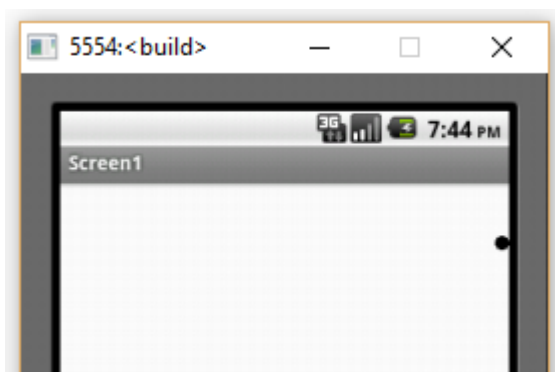


Imagen 147: Otro "frame" de la animación

B. Hacer que una pelota rebote en los bordes (sin requerir componente de tiempo)



Imagen 148: Se pone un objeto Bola al interior del lienzo

En el lienzo, ubique un componente Bola en su interior, renómbrelo a "Bola" y dele valores a los siguientes atributos:

Dirección: Es la dirección hacia donde se dirige en grados. 0 grados es hacia la derecha horizontal (ir al Oriente), 90 grados es hacia arriba vertical (ir al Norte), luego 45 grados sería como ir en dirección Nororiental, 180 grados es ir hacia la izquierda horizontal (ir al Occidente), 270 grados es ir hacia abajo vertical (ir al Sur). Se puede seleccionar entre 0 y 360 grados. En el ejemplo se escogió 45.

Intervalo: Cada cuantos milisegundos cambia de posición la bola.

Velocidad: Cuantos pixeles salta por cada cambio de posición de la bola.

X: Posición inicial en **x** de la bola

Y: Posición inicial en **y** de la bola

Este es el código que debe programarse

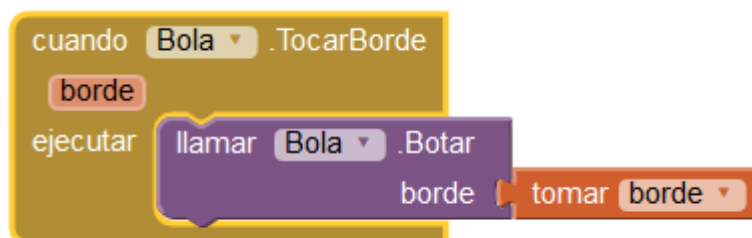


Imagen 149: Código para que la bola rebote en el borde del lienzo

Paso 16. Bases de datos

Si quiere hacer sencillos sistemas para almacenar y recuperar datos, se tiene acceso a un componente de base de datos llamado "TinyDB"

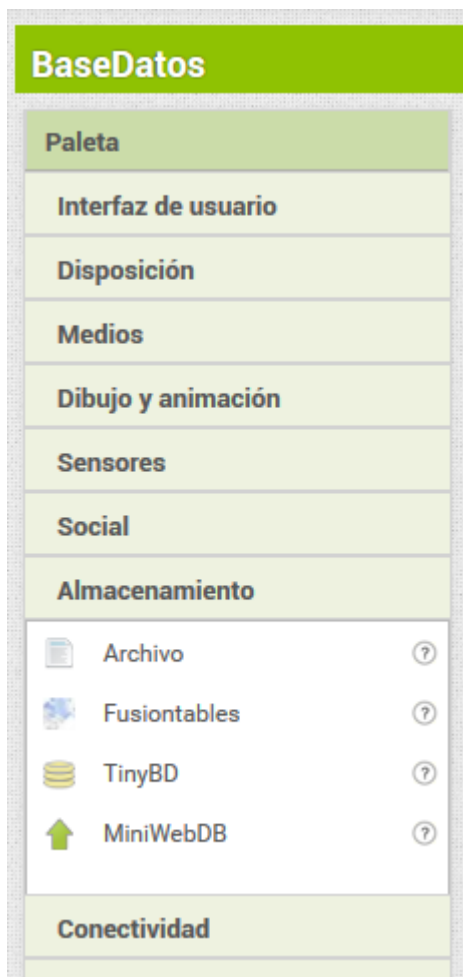


Imagen 150: Se requiere el objeto TinyDB

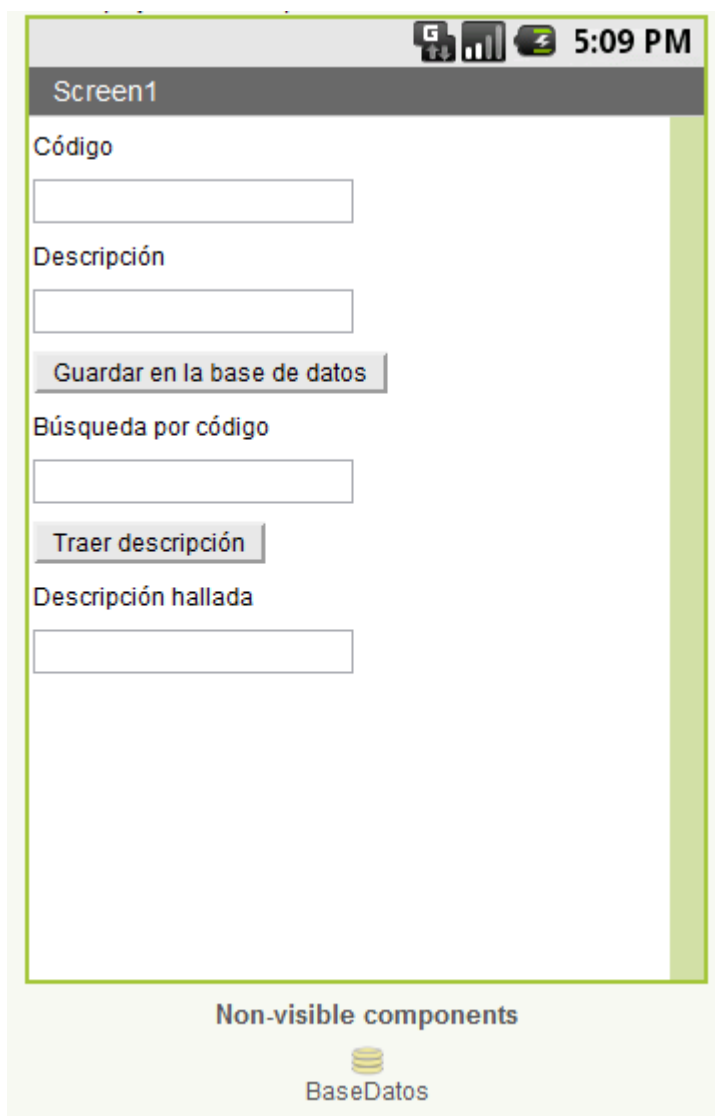


Imagen 151: Formulario para guardar y recuperar datos

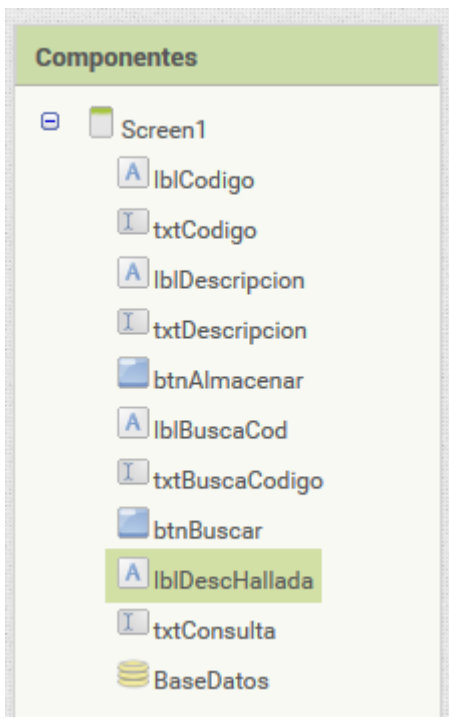


Imagen 152: La lista de componentes u objetos

Para almacenar en la base de datos, se usa la función GuardarValor la cual tiene dos argumentos: una etiqueta que sería una llave primaria y un valorAGuardar que sería el valor a guardar

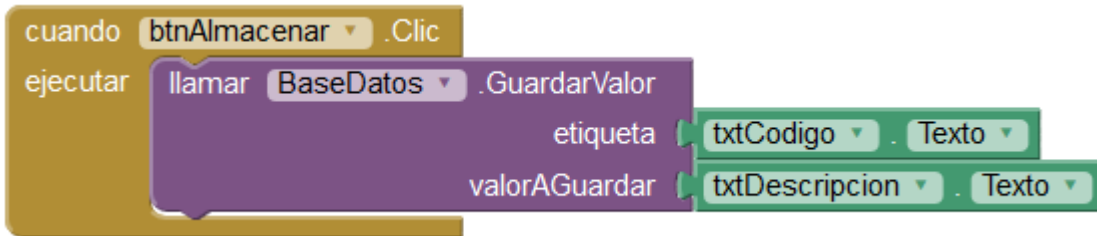


Imagen 153: Guardando registros en la base de datos

Para consultar, se usa la función ObtenerValor que tiene como argumento la llave primaria y retorna el valor almacenado.

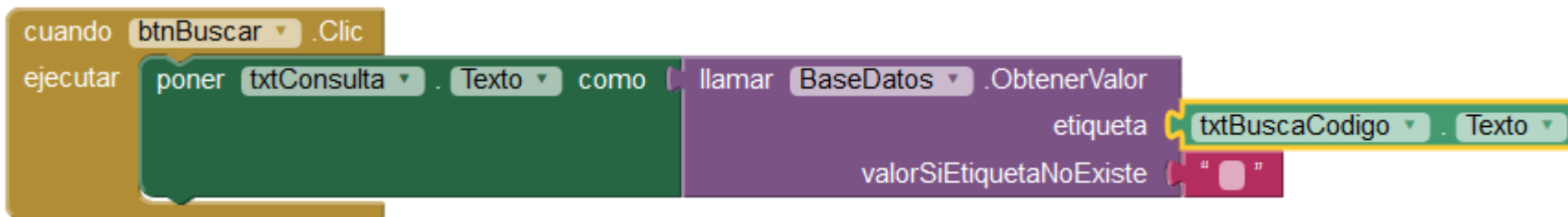


Imagen 154: Restaura el valor almacenado consultando a través de la llave primaria

Paso 17. La capacidad inalámbrica de MIT App Inventor

¿Quiere ejecutar las aplicaciones directamente en un Smartphone o Tablet Android y de manera sencilla? Eso es posible con la nueva funcionalidad wifi de MIT App Inventor. Sólo es seguir los siguientes pasos:

- Instalar la aplicación MIT AI Companion de la Play Store en el Smartphone o Tablet
- Ejecutar ese programa en el Smartphone o Tablet
- En MIT App Inventor decir que se conecta vía Wifi
- MIT App Inventor genera un código que se digita en el Smartphone o Tablet
- Eso es todo, la aplicación corre directamente en el Smartphone o Tablet

A continuación imágenes de este proceso



MIT AI2 Companion 

MIT Center for Mobile Learning

E Todos

INSTALAR



Descargas



13,260 



Educación



Similares

Desarrolla tus propias aplicaciones
Android usando MIT App Inventor 2!

MÁS INFORMACIÓN

Imagen 155: Buscar en la "Play Store" a MIT AI2 Companion

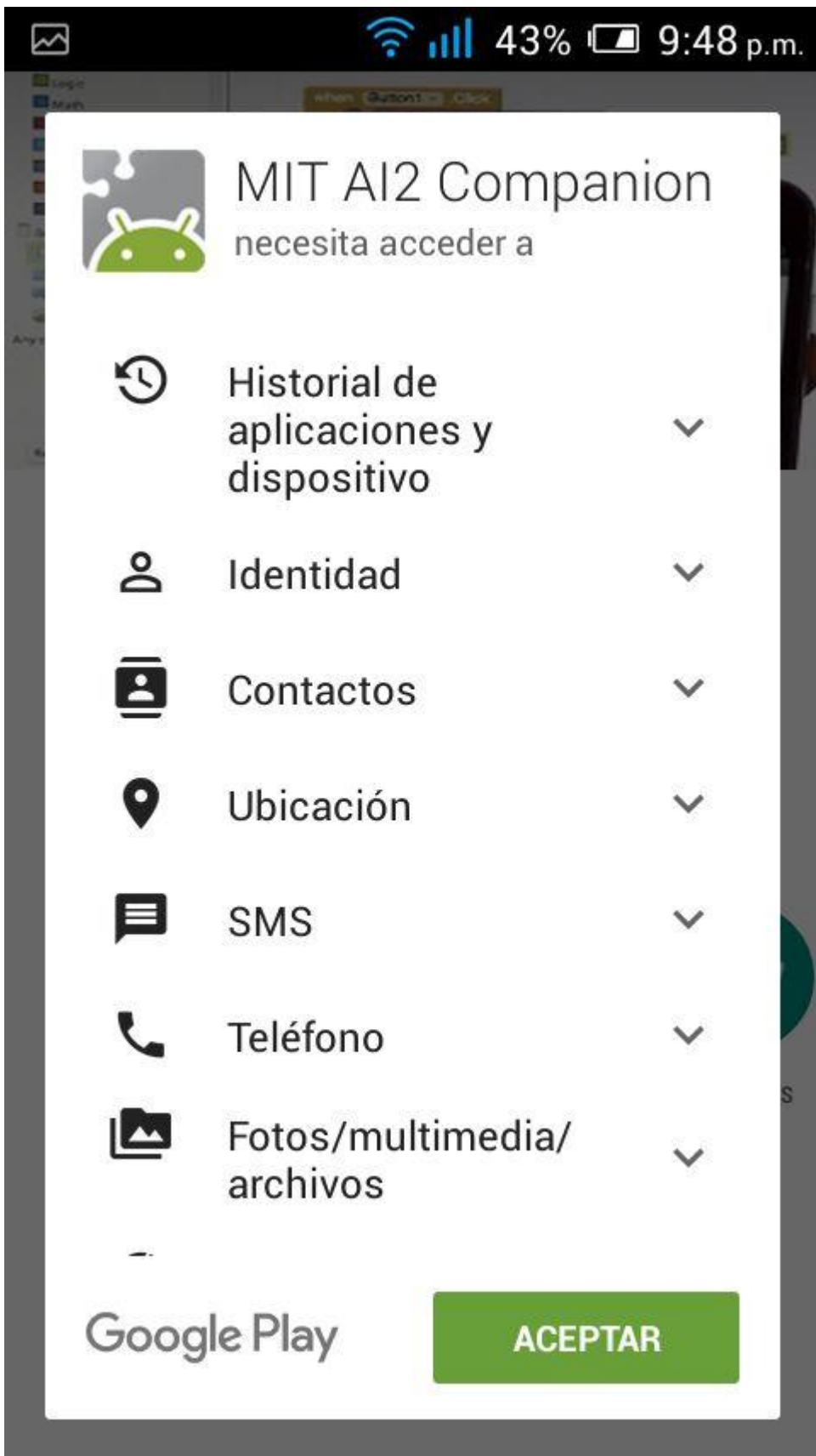


Imagen 156: Se instala la aplicación



Imagen 157: Se ejecuta la aplicación

En el MIT App Inventor se selecciona "Conectar" y luego "AI Companion"

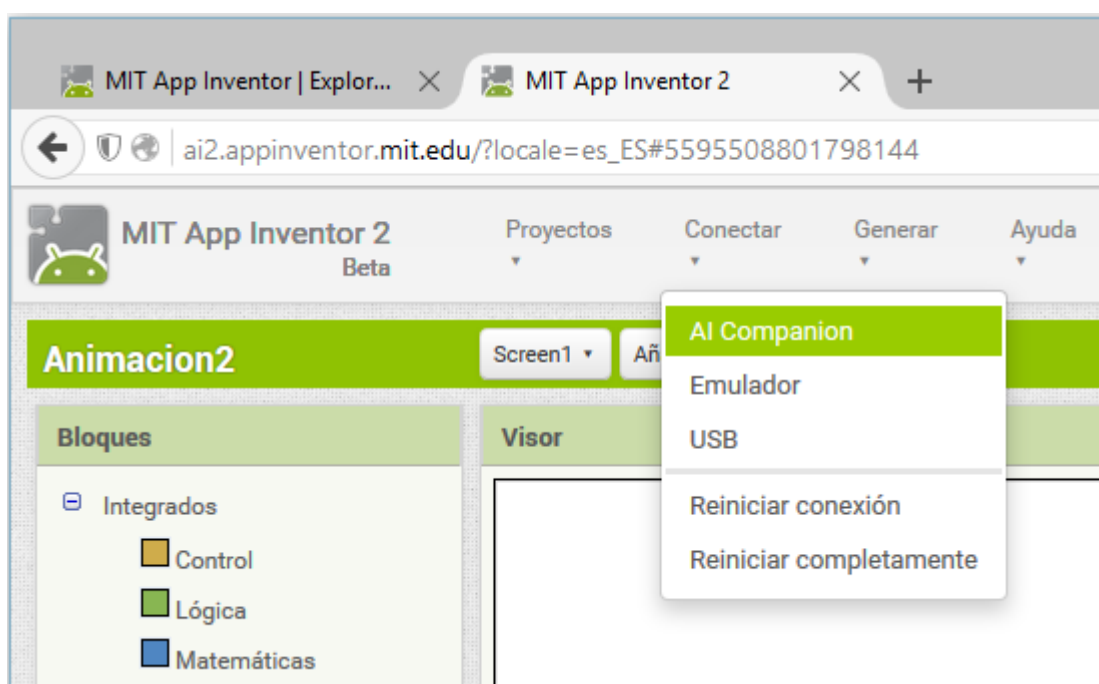


Imagen 158: Se selecciona "Conectar" y luego "AI Companion"

Sale un aviso en el que se digita un código o se escanea un código QR



Imagen 159: En el smartphone o Tablet se digita el código mostrado o se escanea el código QR

Paso 18. Tomando vídeos

Aquí es necesario hacer uso de la capacidad Wifi de MIT App Inventor 2 para probar la capacidad de grabación directa de un Smartphone Android.

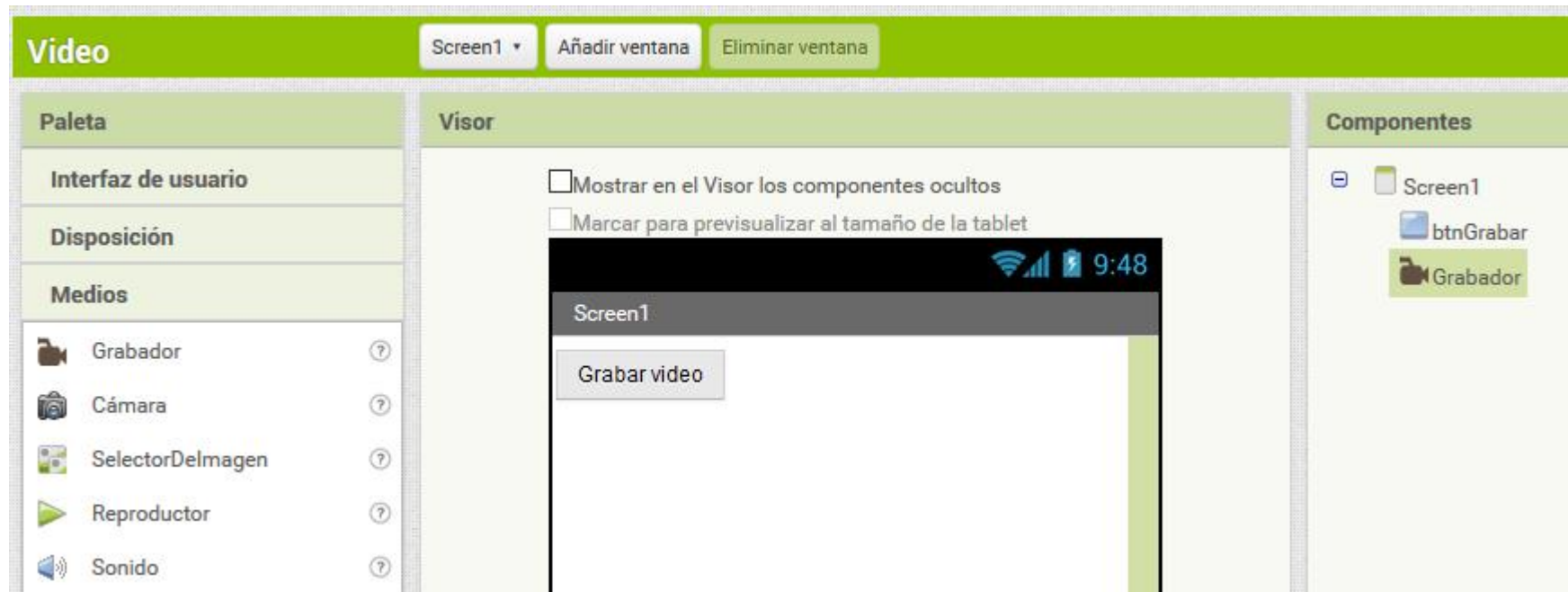


Imagen 160: Se pone un botón y un componente Media llamado "Camcorder"

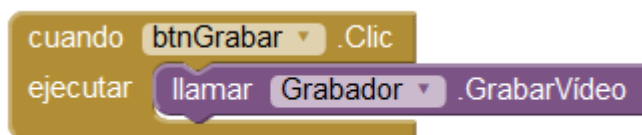


Imagen 161: Ese es el evento a programar

Paso 19. Tomando fotos

Aquí es necesario hacer uso de la capacidad Wifi de MIT App Inventor 2 para probar la capacidad de tomar fotos de un Smartphone Android.



Imagen 162: Se pone un botón y un componente Media llamado "Camera"

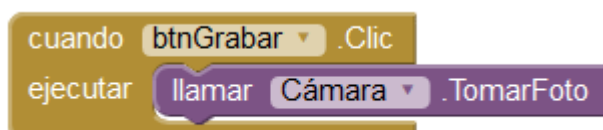


Imagen 163: Ese es el evento a programar

Paso 20. Ver galería de imágenes

Se toma el control "ImagePicker" que es similar a un botón, al dar clic sobre este, se accede a la galería de imágenes que tiene el Smartphone.



Imagen 164: SelectorDeImagen es el control a usar

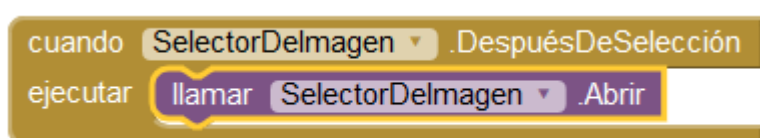
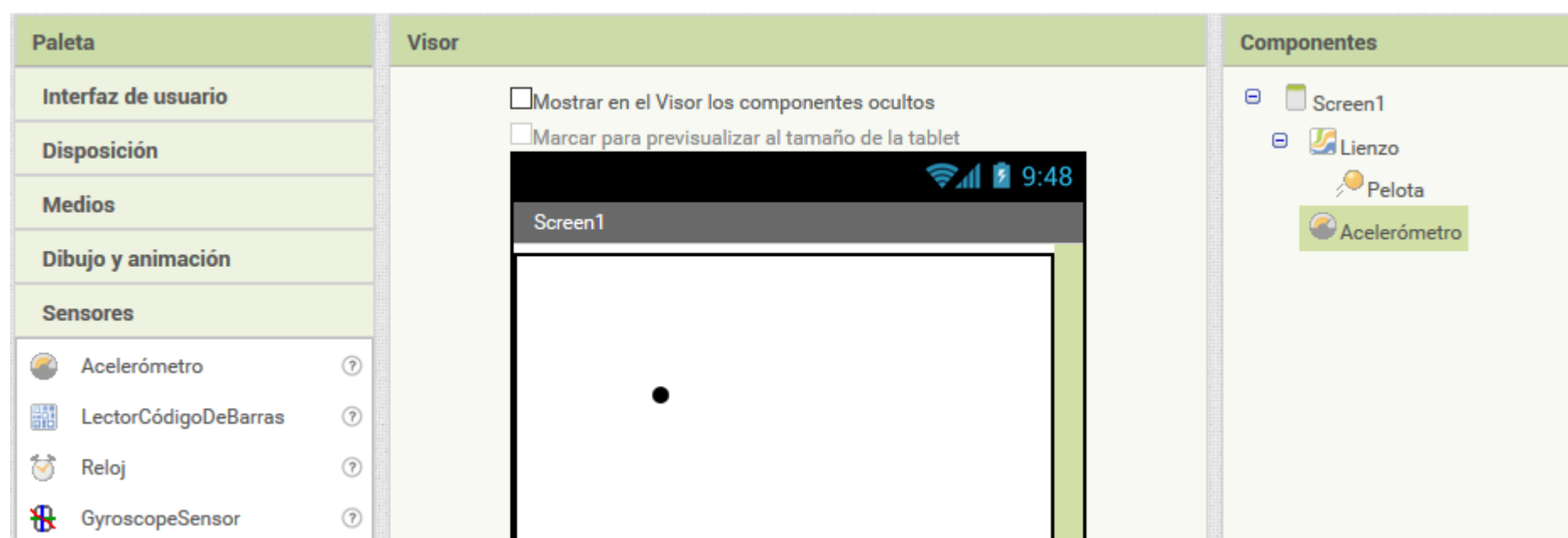


Imagen 165: Al presionar se muestra la galería de imágenes

Paso 21. Usando el acelerómetro

En el ejemplo, se va a mover una bolita según como mueva el Smartphone



Se tiene un componente Lienzo y en el interior un componente Pelota, y se agrega el componente Acelerómetro.

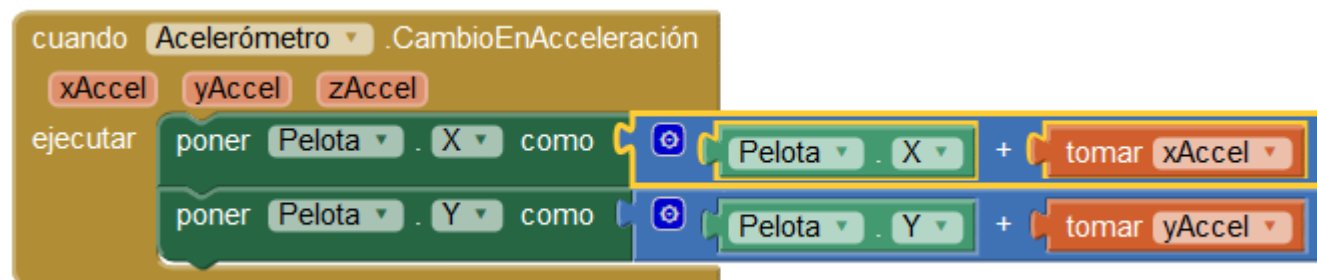


Imagen 166: Usando el acelerómetro para mover una figura animada.

Paso 22. Organización en pantalla

Los componentes visuales pueden ordenarse en la pantalla de tres formas: Horizontal, en forma de tabla y vertical.

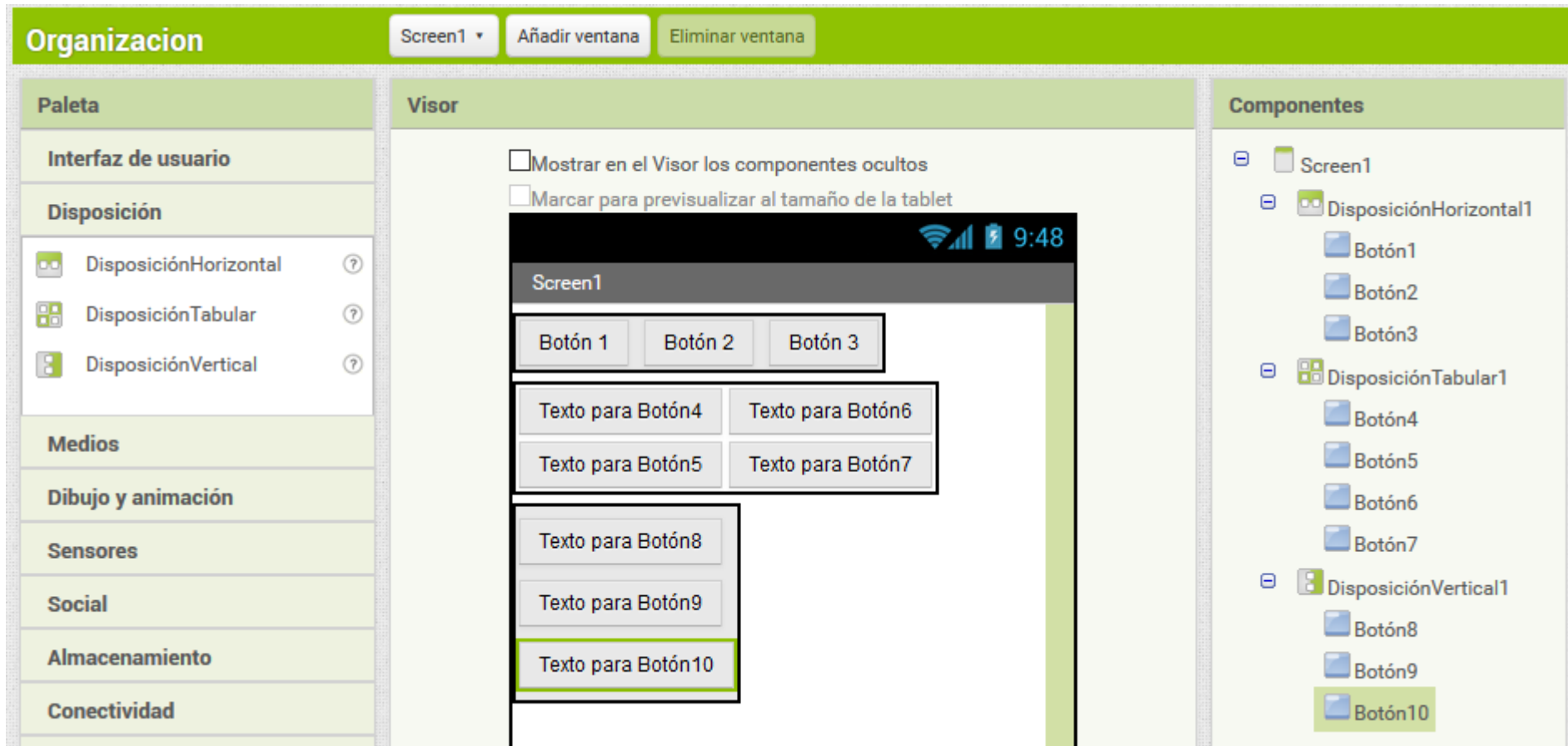


Imagen 167: Uso de las tres formas de ordenar en pantalla