

CREA TUS APPS CON MIT APP INVENTOR

MATERIALES DE INICIO A LA PROGRAMACION EN ANDROID



fundación **esplai**
ciudadanía comprometida



Con el apoyo de



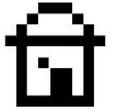
Microsoft

Financiado por



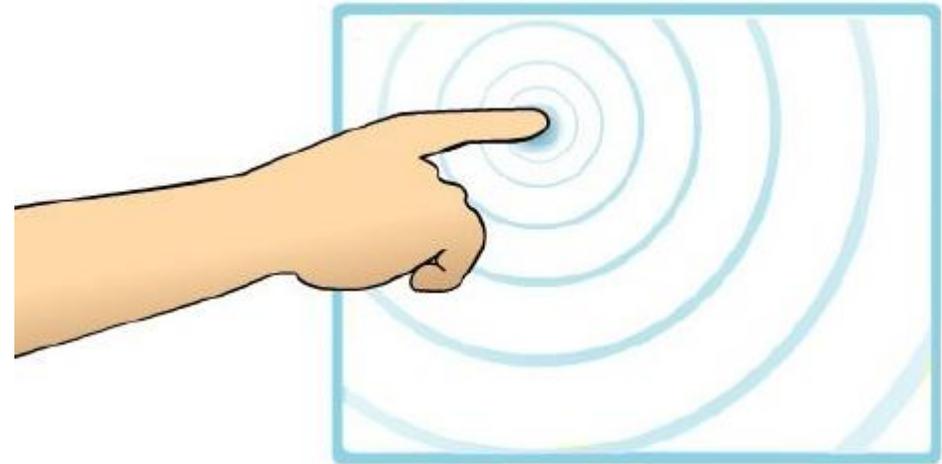
✓ POR COLABORAR
OTROS FINES DE INTERÉS SOCIAL

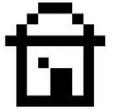
1. Qué es App Inventor
2. Entorno web
3. Probar las Apps
4. Primera App: sumas
5. Segunda App: par o impar
6. Tercera App: lienzo
7. Cuarta App: app multiventana
8. Quinta App: nivel de burbuja



1. QUE ES APP INVENTOR

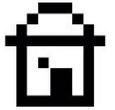
Hoy en día, el uso de los dispositivos móviles es de lo más habitual, tanto para uso personal como profesional. Normalmente la mayoría de estos dispositivos disponen de tecnología táctil para hacer más sencillo e intuitivo su uso, especialmente a personas no familiarizadas con la tecnología.





MIT App Inventor es una **herramienta web de programación visual** y con ella vas a poder crear tus propias Apps para dispositivos Android. Con **App Inventor** no necesitas tener unos extensos conocimientos sobre sintaxis de programación, ya que mediante la unión de bloques vas a poder dotar de instrucciones a los componentes que vayan a formar tus Apps, creando desde aplicaciones muy simples hasta aplicaciones mucho mas complejas.



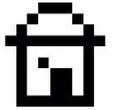


CREAR UNA CUENTA DE APP INVENTOR

App Inventor es un producto desarrollado por el **Instituto Tecnológico de Massachusetts (MIT)** junto con **Google**, de manera que como requisito se debe disponer de una cuenta de Google para acceder a la web de MIT App Inventor.

Seguidamente podrás ver paso a paso como crear un perfil de App Inventor y así comenzar a programar.





Paso 1: Acceder a la web MIT App Inventor

Accede a la web oficial mediante este enlace: <http://appinventor.mit.edu/>

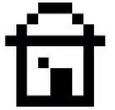
Paso 2: Crear un acceso a la web

Una vez tengas abierta la web, verás en la barra superior un icono de color naranja donde dice **Create apps!**, haz clic en él y asocia tu cuenta de Google para acceder a la plataforma y comenzar a programar. Más adelante verás cómo cambiar el idioma. Si no tienes una cuenta de Google, puedes crearla en: <https://accounts.google.com/SignUp?hl=es>



Paso 3: Comenzar a trabajar

Una vez hayas introducido tu cuenta de Google, te aparecerá un texto con el Acuerdo de Licencia de Uso, y al pulsar aceptar ya tendrás acceso a la interfaz de programación como se muestra en la siguiente diapositiva.



To use App Inventor for Android, you must accept the following terms of service.

Terms of Service

MIT App Inventor Privacy Policy and Terms of Use

MIT Center for Mobile Learning

Welcome to MIT's Center for Mobile Learning's App Inventor website (the "Site"). The Site runs on Google's App Engine service. You must read and agree to these Terms of Service and Privacy Policy (collectively, the "Terms") prior to using any portion of this Site. These Terms are an agreement between you and the Massachusetts Institute of Technology. If you do not understand or do not agree to be bound by these Terms, please immediately exit this Site.

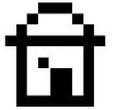
MIT reserves the right to modify these Terms at any time and will publish notice of any such modifications online on this page for a reasonable period of time following such modifications, and by changing the effective date of these Terms. By continuing to access the Site after notice of such changes have been posted, you signify your agreement to be bound by them. Be sure to return to this page periodically to ensure familiarity with the most current version of these Terms.

Description of MIT App Inventor

From this Site you can access MIT App Inventor, which lets you develop applications for Android devices using a web browser and either a connected phone or emulator. You can also use the Site to store your work and keep track of your projects. App Inventor was originally developed by Google. The Site also includes documentation and educational content, and this is being licensed to you under the Creative Commons Attribution 4.0 International license ([CC BY 4.0](#)).

Account Required for Use of MIT App Inventor

I accept the terms of service!



MIT App Inventor 2
Beta

Projects ▾ Connect ▾ Build ▾ Help ▾

My Projects Gallery Guide Report an Issue English ▾ fundacionesplai@gmail.com ▾

Start new project Delete Project Publish to Gallery

My Projects

Name	Date Created	Date Modified ▾	Published
------	--------------	-----------------	-----------

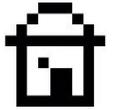
Welcome to App Inventor 2!



You do not have any projects in App Inventor 2. To learn how to use App Inventor, click the "Guide" link at the top of the window; or to start your first project, click the "Start New Project" button at the upper left of the window.

Happy Inventing!

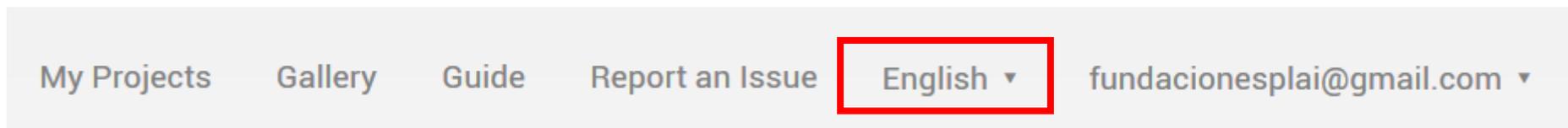
[Privacy Policy and Terms of Use](#)



CAMBIAR EL IDIOMA

La primera vez que abres la web aparece un cuadro de texto en el centro informándote de que no tienes proyectos, animándote a crear nuevos.

La web de App Inventor de manera predeterminada se muestra en inglés, pero en la barra de herramientas superior podrás ver una opción desplegable con todos los idiomas disponibles. Haz clic en la pestaña y selecciona **Español**.





2. ENTORNO WEB

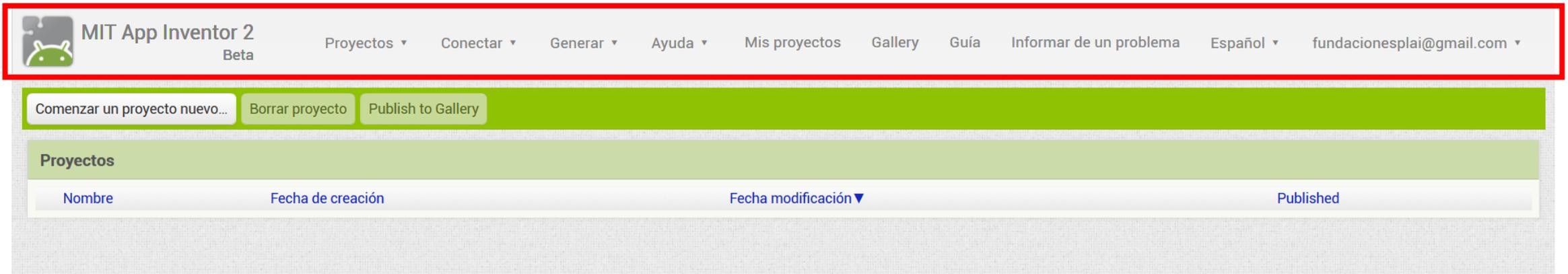
App Inventor es un entorno de **programación visual**, que permite crear Apps usando **objetos** e **instrucciones** en forma de bloques. Estas **instrucciones** proporcionarán funcionalidades a los **objetos** o componentes de la App para que hagan una u otra cosa.

En las siguientes diapositivas podrás ver las diferentes partes del entorno de trabajo.

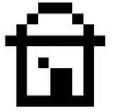
- Barra de herramientas
- Barra de la App abierta
- Paleta
- Visor
- Componentes
- Media
- Propiedades



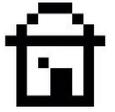
BARRA DE HERRAMIENTAS



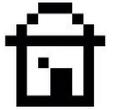
La **barra de herramientas** (marcada en rojo) tiene diferentes opciones que te permitirán acceder a la creación y edición de Apps.



- **Proyectos:** crear, editar, guardar, exportar, importar nuestros proyectos.
- **Conectar:** probar el funcionamiento de nuestra App, ya sea en el emulador propio de MIT App Inventor o bien en un dispositivo móvil. Con esta opción podrás ir haciendo pruebas a medida que vas programando la App.
- **Generar:** una vez tengas la App definitiva, vas a poder exportarla al PC o bien al móvil directamente mediante la tecnología QR, para de este modo hacer un uso real de lo que has creado.
- **Ayuda:** acceso a todo tipo de información propia de App Inventor, desde tutoriales hasta foros de la comunidad.
- **Mis proyectos:** acceso a la librería en la que se almacenan todos los proyectos que tienes guardados en tu perfil.



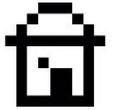
- **Galería:** acceso a las aplicaciones compartidas en la comunidad, tutoriales y demás documentos compartidos por otros usuarios de la comunidad de MIT App Inventor.
- **Guía:** acceso a documentación de soporte oficial que te ayudará con dudas que se puedan plantear.
- **Idioma:** podrás cambiar el idioma de la interfaz web tal y como has visto anteriormente.



BARRA DE LA APP ABIERTA

Esta barra de herramientas es la que ofrece las opciones que hacen referencia a la app que actualmente se encuentra abierta.

- **Ventanas:** permite cambiar la ventana del programa con la cual quieres trabajar en un momento concreto, **ya que una app puede disponer de varias ventanas.**
- **Añadir ventanas:** permite añadir nuevas ventanas a la app que estás desarrollando.
- **Eliminar ventanas:** permite eliminar las ventanas del proyecto en el que trabajamos.



MIT App Inventor 2 Beta

Proyectos ▾ Conectar ▾ Generar ▾ Ayuda ▾

Mis proyectos Gallery Guía Informar de un problema Español ▾ fundacionesplai@gmail.com ▾

calculadora Screen1 ▾ Añadir ventana Eliminar ventana Diseñador Bloques

Paleta

Interfaz de usuario

- Botón
- CasillaDeVerificación
- SelectorDeFecha
- Imagen
- Etiqueta
- SelectorDeLista
- VisorDeLista
- Notificador
- CampoDeContraseña
- Deslizador
- Desplegable
- CampoDeTexto
- SelectorDeHora
- VisorWeb

Disposición

Medios

Dibujo y animación

Visor

Mostrar en el Visor los componentes ocultos

Marcar para previsualizar al tamaño de la tablet

Screen1

Screen1

9:48

Screen1

Componentes

- Screen1

Cambiar nombre Borrar

Medios

Subir archivo...

Propiedades

Screen1

PantallaAcercaDe

DispHorizontal

Izquierda : 1 ▾

DispVertical

Arriba : 1 ▾

AppName

calculadora

ColorDeFondo

Blanco

ImagenDeFondo

Ninguno...

AnimaciónCierreDePantalla

Por defecto ▾

Icono

Ninguno...

AnimaciónAlAbrirPantalla

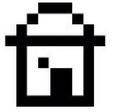
Por defecto ▾

OrientaciónDeLaPantalla

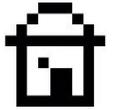
Sin especificar ▾

Enrollable

ShowListsAsJson



- **Ventanas** – Esta pestaña da la opción de cambiar la ventana del programa con la cual quieres trabajar en un omento concreto, ya que una app puede disponer de varias ventanas.
- **Añadir ventanas** – Esta opción permite añadir nuevas ventanas a la app que estas desarrollando.
- **Eliminar ventanas** – Esta opción permite eliminar las ventanas del proyecto.
- **Diseñador:** permite acceder a la ventana de diseño para colocar los objetos que forman la interfaz de nuestra App y hacer la composición del aspecto, los botones y el resto de elementos que tendrá.
- **Bloques:** permite añadir instrucciones en forma de bloques a los elementos que se han agregado en el apartado de **Diseñador**. Desde aquí programaremos el funcionamiento de nuestra App, añadiendo funcionalidades a los objetos que hemos colocado.

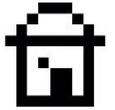


PALETA

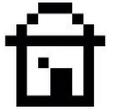
Esta sección del entorno situada en el lateral izquierdo de la interfaz de App Inventor, contiene todos los **elementos visuales o de otro tipo, que pueden ser agregados y tratados en una App.**

Al pulsar en cada una de las opciones se despliega, abajo, el menú de elementos correspondientes a esa categoría.

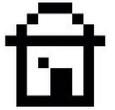
Paleta
Interfaz de usuario
Disposición
Medios
Dibujo y animación
Sensores
Social
Almacenamiento
Conectividad
LEGO® MINDSTORMS®
Experimental
Extension



- **Interfaz de usuario:** todos los elementos interactivos que se pueden agregar a una App, como por ejemplo botones, checkboxes (cuadros de selección), barras de desplazamiento, imágenes, listas, desplegados, campos de texto...
- **Disposición:** permite añadir capas “no visibles” con el fin de organizar los elementos dentro de la pantalla. Pueden ser dispuestos de manera vertical, horizontal, dentro de tablas...
- **Medios:** para añadir reproductores de música y video hasta galerías de imágenes. Este apartado contiene “**Funciones interactivas predeterminadas**”, es decir, listas para ser agregadas y utilizadas.
- **Dibujo y animación:** una serie de objetos visuales e interactivos con los cuales podrás dibujar, interactuar entre objetos...

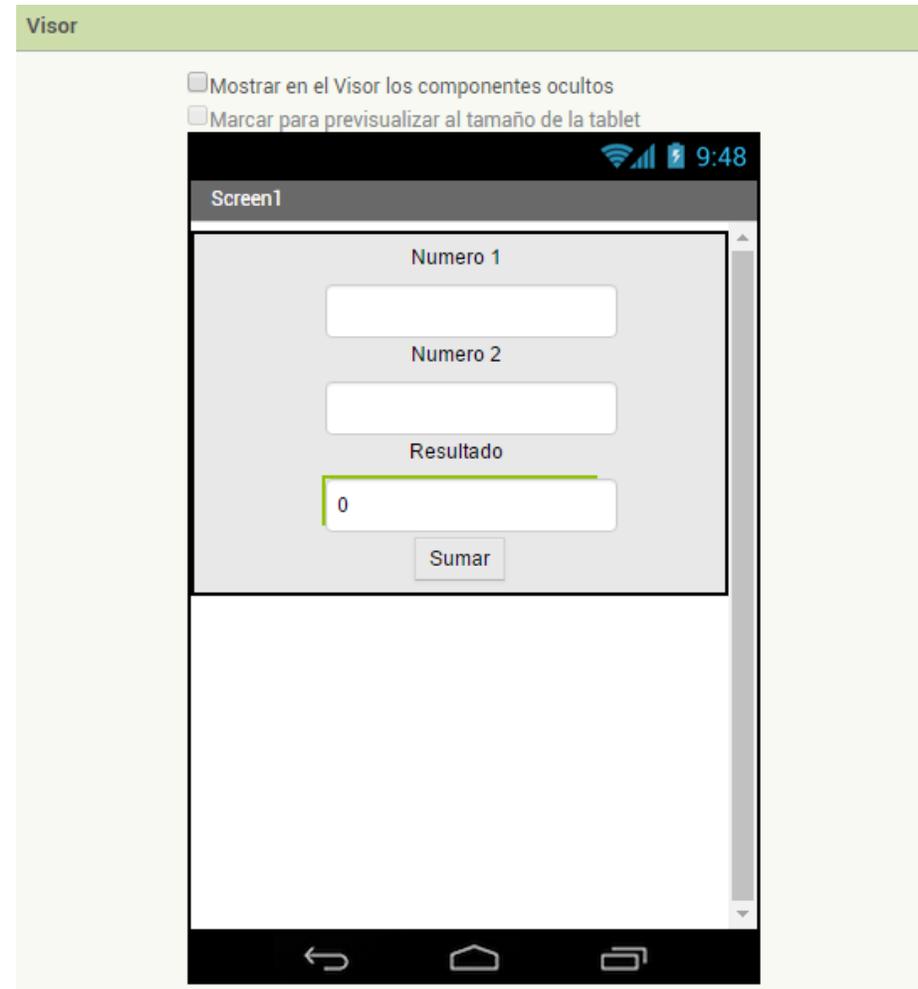


- **Sensores:** este apartado es bastante relevante, ya que es con el que podrás agregar a tu app funcionalidades que un dispositivo móvil incorpora hoy en día de manera habitual, tales como; acelerómetro, giroscopio, podómetro, brújula...
- **Social:** una app también puede incorporar herramientas clásicas como; SMS, e-mail, navegador web y redes sociales, agenda de contactos...
- **Almacenamiento:** como en todo tipo de programación, el hecho de reutilizar datos que obtenemos o que se encuentran almacenados es muy habitual y necesario. Con este apartado vas a poder generar tu base de datos para incluirla en tu app y utilizar archivos con información necesaria para el buen funcionamiento de tu App.
- **Conectividad:** otra gran característica de las app móvil es la capacidad de comunicarse con otros dispositivos del entorno mediante tecnologías inalámbricas como Bluetooth ya incorporadas en los terminales.



VISOR

Este apartado del entorno muestra lo que se verá en la App una vez la instales o emules en un dispositivo. Mediante los componentes de la sección **Paleta** que iras arrastrando al visor, se ira generando el aspecto visual de tu App.

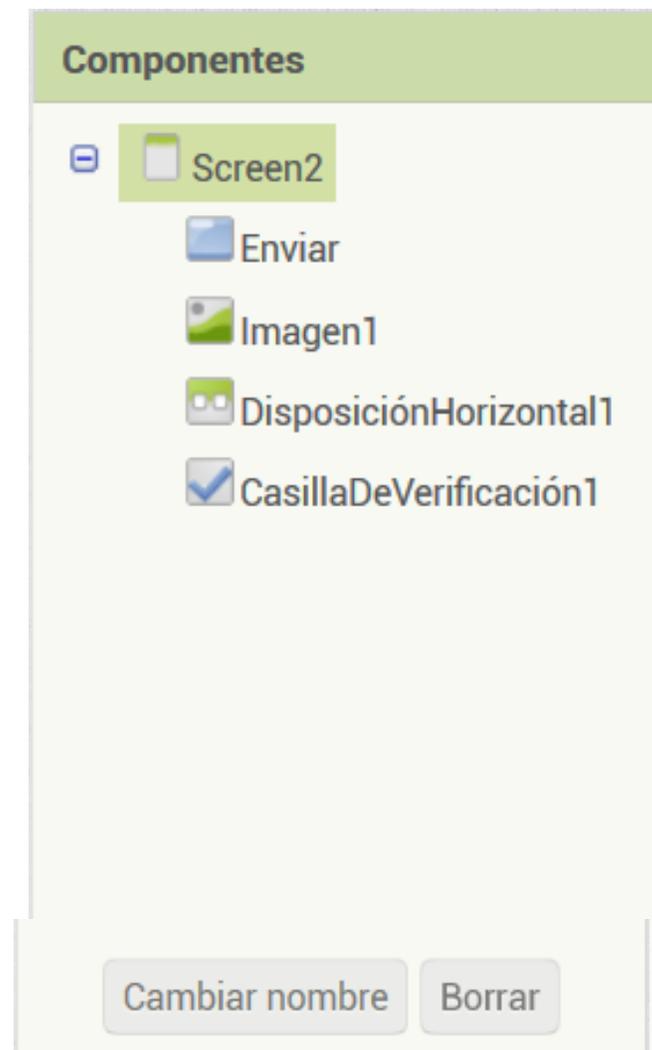




COMPONENTES

Cada vez que se añada un componente al visor de tu App, se irá generando una lista de éstos, de manera que podrás identificar rápidamente con qué objetos estás trabajando en todo momento.

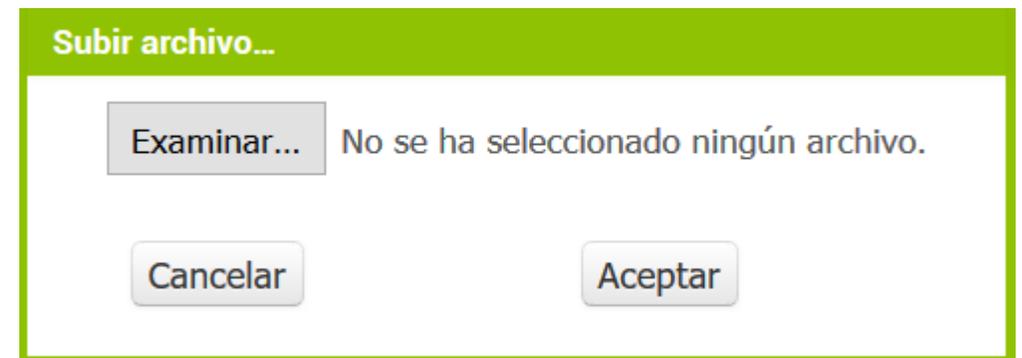
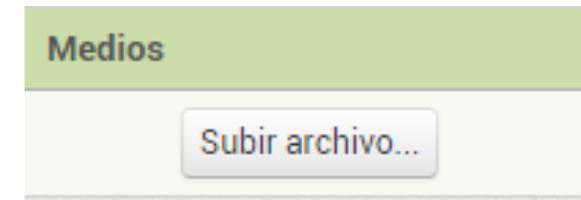
Para renombrar o eliminar un elemento no deseado, lo tienes que hacer mediante esta sección del entorno.





MEDIOS

Cuando quieras agregar archivos digitales desde el ordenador, tales como música, fotos o videos a la App, lo vas a tener que hacer desde este apartado. El botón **Subir archivo** permite acceder a los archivos que se encuentran almacenados dentro del ordenador desde el que se está ejecutando la herramienta web.





PROPIEDADES

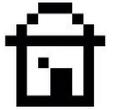
Al igual que cada vez que se añade un elemento al Visor, éste se añade a una lista dentro del apartado Componentes de manera automática, también se va a generar la sección Propiedades donde se van a poder modificar ciertos parámetros del componente del visor seleccionado: el aspecto de un botón, de una imagen...

The screenshot displays a web development interface with two main panels: 'Componentes' and 'Propiedades'.

Componentes: A list of components is shown, including 'Screen2', 'Enviar', 'Imagen1' (highlighted), 'DisposiciónHorizontal1', and 'CasillaDeVerificación1'.

Propiedades: The properties for the selected 'Imagen1' component are listed:

- Alto:** Automático...
- Ancho:** Automático...
- Foto:** Ninguno...
- ÁnguloRotación:** 0.0
- EscalarFotoAlTamañoMáximo:**
- Visible:**

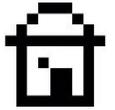


BLOQUES

Como ya sabes, por un lado deberás desarrollar visualmente tu App agregando y distribuyendo los componentes dentro de la pantalla, pero también deberás **programar con bloques el comportamiento de estos componentes**. Para ello, como ya has visto antes, está la sección de **Bloques**.

Cada objeto o componente tiene una serie de instrucciones organizadas por secciones de manera predeterminada.

En la siguiente diapositiva podrás ver el aspecto de la sección de **Bloques**.



MIT App Inventor 2 Beta

Proyectos ▾ Conectar ▾ Generar ▾ Ayuda ▾

Mis proyectos Gallery Guía Informar de un problema Español ▾ fundacionesplai@gmail.com ▾

calculadora Screen2 ▾ Añadir ventana Eliminar ventana Diseñador Bloques

Bloques

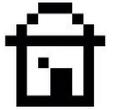
- Integrados
 - Control
 - Lógica
 - Matemáticas
 - Texto
 - Listas
 - Colores
 - Variables
 - Procedimientos
- Screen2
 - Enviar
 - Imagen1
 - DisposiciónHorizontal1
 - CasillaDeVerificación1
- Cualquier componente

Cambiar nombre Borrar

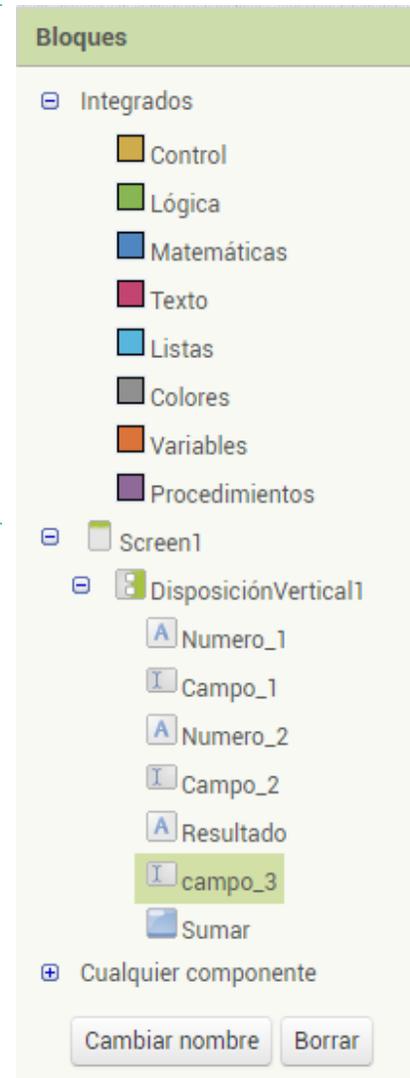
Medios

Visor

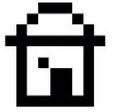
```
si entonces
por cada número desde 1 hasta 5 en incrementos de 1 ejecuta
por cada elemento en la lista ejecutar
mientras comprobar ejecutar
si entonces si no
ejecutar avisos resultado
```



En el apartado de bloques, encontrarás dos partes. La primera contiene las distintas secciones de bloques con instrucciones generales que normalmente se pueden aplicar a cualquier componente. Digamos que son las **instrucciones comunes**.



En el segundo apartado encontrarás que aparece el nombre de los componentes añadidos en el visor. Aquí se da la oportunidad de trabajar con **instrucciones propias** para este tipo de componentes.

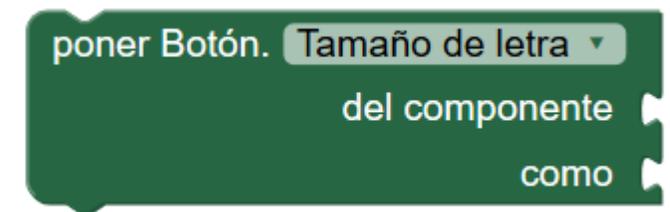


Algunos ejemplos:

En el apartado de **instrucciones comunes**, si haces clic en la sección de **control** verás que hay bloques tales como (Si - Entonces) que es del tipo control de condiciones y puede ser aplicado a la gran mayoría de bloques.



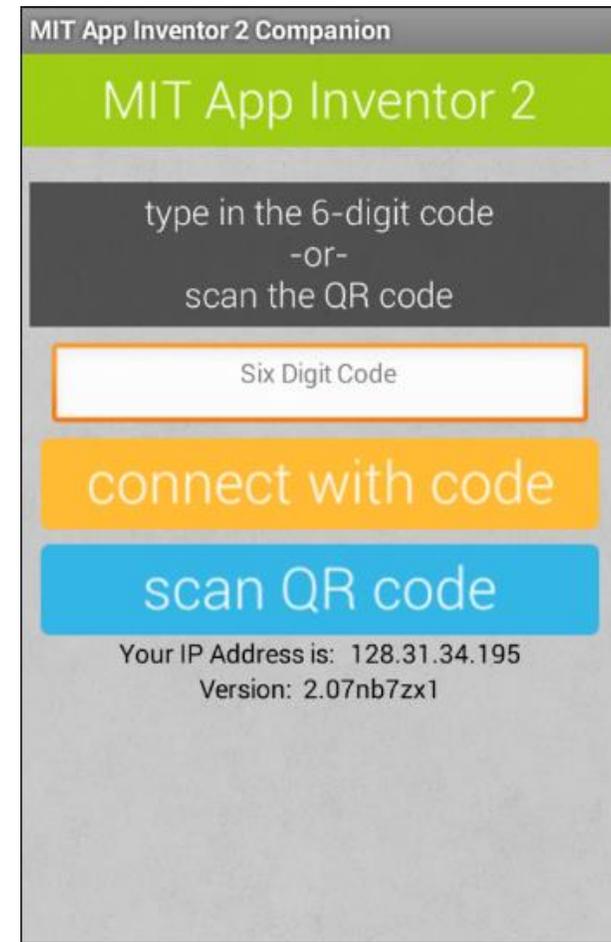
En el apartado de **instrucciones específicas** encontrarás bloques como el siguiente que te permitirá cambiar el tamaño de letra de un determinado componente como un botón, por ejemplo, a un determinado tamaño.

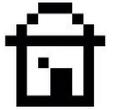




3. PROBAR LAS APPS

A medida que vas desarrollando una App seguramente irás haciendo pruebas para ir corrigiendo los errores. La manera más sencilla de ejecutar estas pruebas es mediante la pestaña **Conectar** situada en la barra de herramientas. Pero antes de nada vas a tener que instalar MIT App Inventor 2 Companion en tu móvil.





MIT APP INVENTOR 2 COMPANION

Desde el mismo dispositivo móvil que vayas a utilizar para ejecutar la app, debes buscar mediante [Google Play Store](#) la app [MIT App Inventor 2 Companion](#).

Una vez se haya descargado e instalado la app, es muy importante estar conectado con este dispositivo móvil a la misma red de Internet a la que está conectado el PC en el cual se está ejecutando la interfaz web de MIT App Inventor.



MIT AI2 Companion

MIT Center for Mobile Learning Educación ★★★★★ 15.845

3 PEGI 3

i Esta aplicación es compatible con todos tus dispositivos.

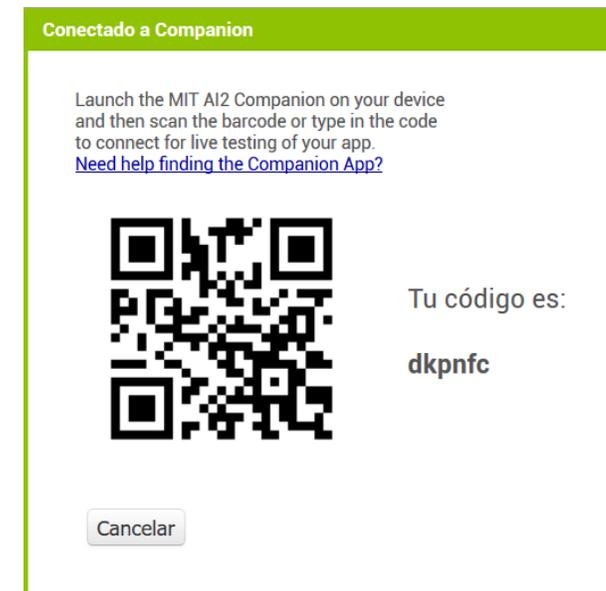
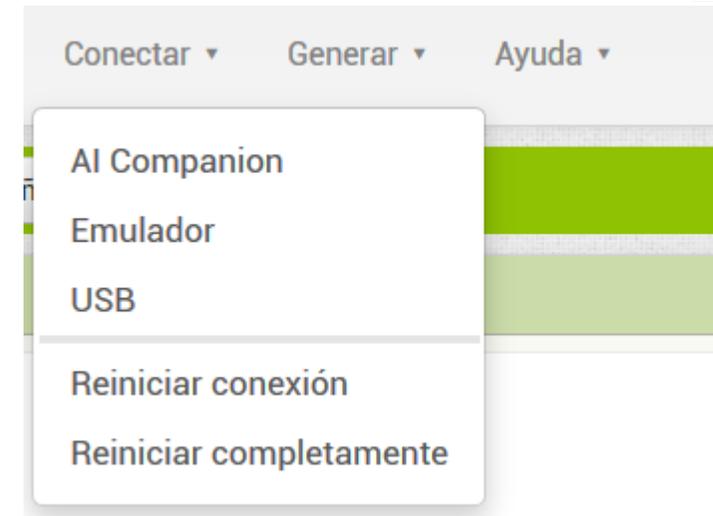
Instalada

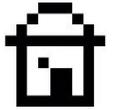


Una vez instalado, debes ir a la interfaz web y dentro de la pestaña **Conectar** situada en la barra de herramientas, vas a escoger la opción que dice **AI Companion**.

Una vez ejecutado **AI Companion** en el ordenador verás que se abre una ventana con un código de 6 caracteres.

Seguidamente ejecuta **AI Companion** en tu dispositivo móvil.





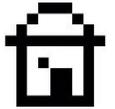
Una vez abierta la app de **AI companion** en tu dispositivo móvil, deberás escribir los 6 caracteres que aparecieron en el ordenador, dentro de la caja de texto que dice **Six Digit Code** y seguidamente presiona la opción **connect with code**.

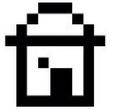
De este modo se ejecutará de manera automática la aplicación que estás desarrollando dentro de tu dispositivo móvil para que veas el resultado de tu trabajo.



APP INVENTOR 2 CLIENT

Otro modo de realizar pruebas a medida que se va creando la App es con un **emulador** dentro del ordenador. Para ello deberás instalar un complemento llamado **App Inventor 2 Client**. Seguidamente verás paso a paso cómo realizar esta instalación.





Primero vas a descargar el software cliente **App Inventor 2 Client** y posteriormente lo vas a instalar, siguiendo las instrucciones del propio instalador.

Accedes a la descarga mediante este enlace:

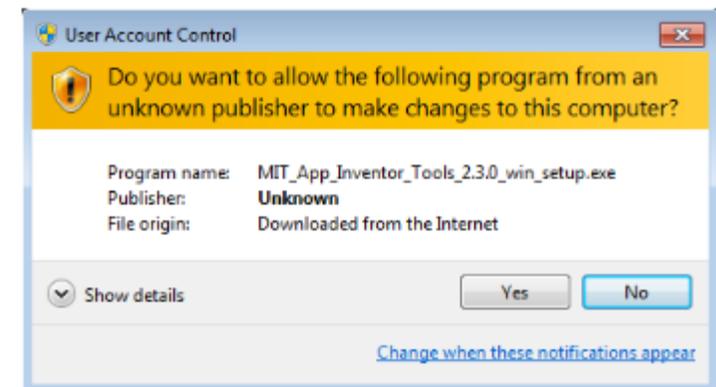
<http://appinventor.mit.edu/explore/ai2/windows.html>

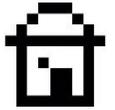
Installing the App Inventor Setup software package

You must perform the installation from an account that has admini

If you have installed a previous version of the App Inventor 2 setup to Follow the instructions at [How to Update the App Inventor Setup Softw](#)

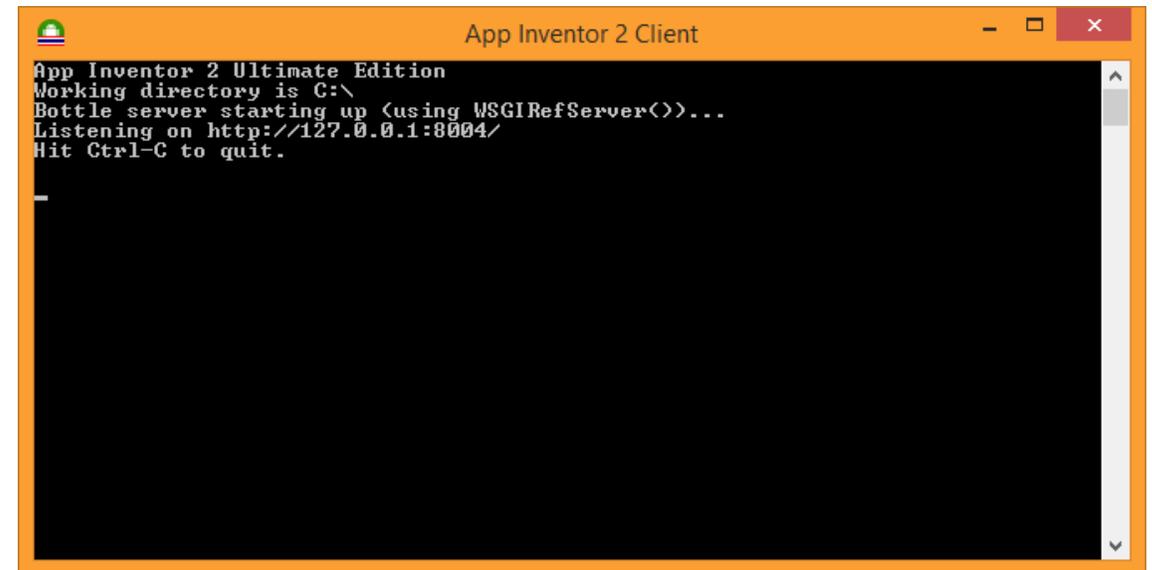
1. Download the installer.
2. Locate the file **MIT_Appinventor_Tools_2.3.0 (~80 MB)** in your on how your browser is configured.
3. Open the file.
4. Click through the steps of the installer. Do not change the install: drivers later. The directory will differ depending on your version of
5. You may be asked if you want to allow a program from an **unkno**

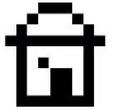




Una vez se haya instalado simplemente tienes que ejecutarlo y dejarlo abierto. Haz doble clic en el icono que aparecerá en tu escritorio.

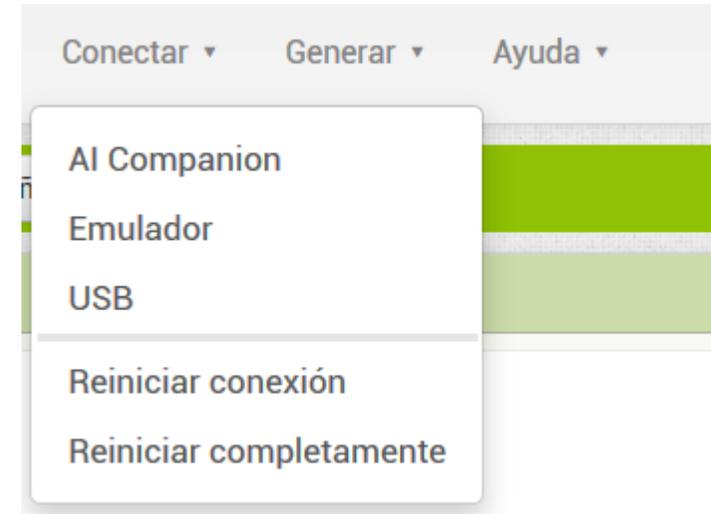
Aparecerá la ventana que tienes a la derecha. **Déjala abierta mientras quieras probar tu App en el emulador.** Puedes minimizarla, pero no cerrarla.

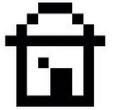




Una vez ejecutado el cliente, dirígete al interfaz web de App Inventor y busca la pestaña **Connect**.

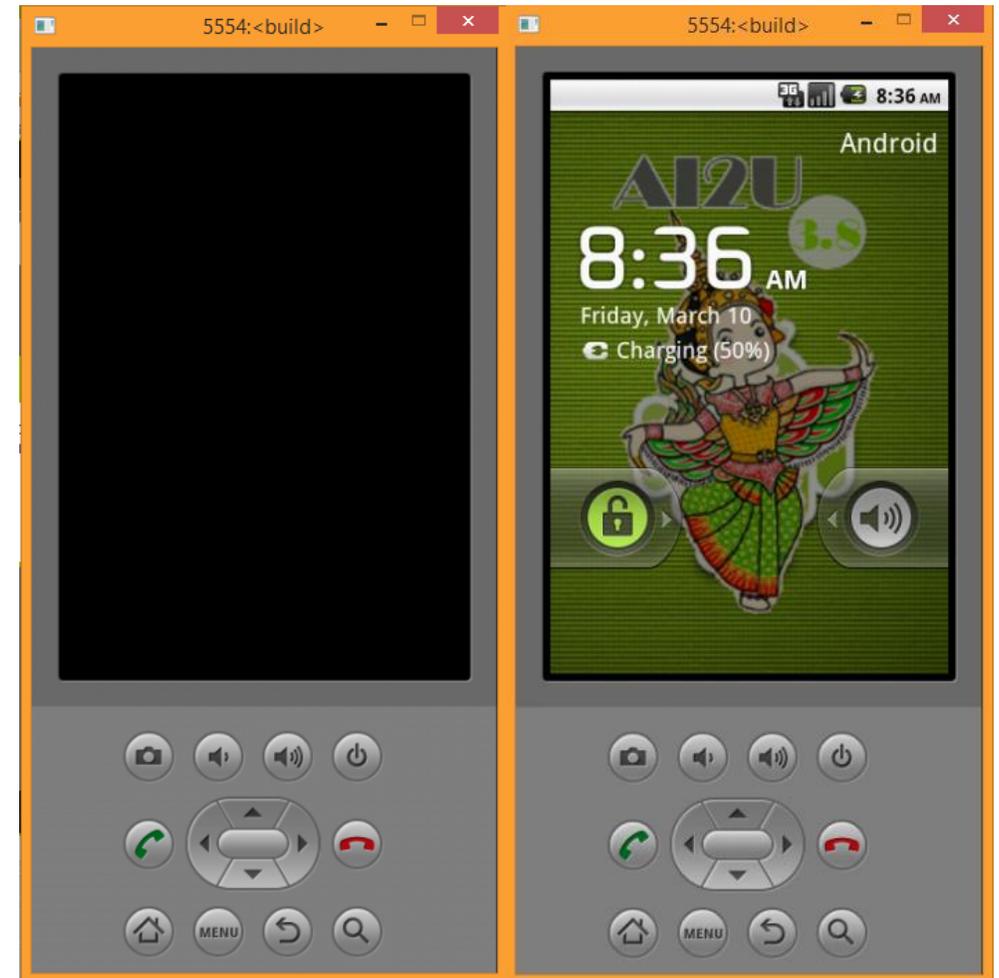
Al hacer clic en dicha pestaña se van a desplegar una serie de opciones, escoge la que dice **Emulador** y espera a que se inicie la interfaz Android simulando un teléfono.

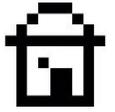




Una vez se ejecute el emulador, solo debes esperar hasta que se inicie la App. En el caso de que no se abra automáticamente, dirígete al menú del emulador (tal y como lo harías en un móvil) y ejecútala manualmente.

Dependiendo del PC donde se ejecute el emulador, puede tardar más o menos, paciencia y no insistir. 😊



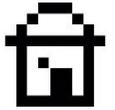


4. PRIMERA APP: SUMAS

Ya es hora de comenzar a programar.
Vas a crear una primera aplicación muy sencilla que consiste en. **sumar dos números y mostrar el resultado**

Para realizar esta práctica, vas a tener que seguir los pasos que se muestran a continuación.

$$\begin{array}{r} 5 \\ +8 \\ \hline 13 \end{array}$$

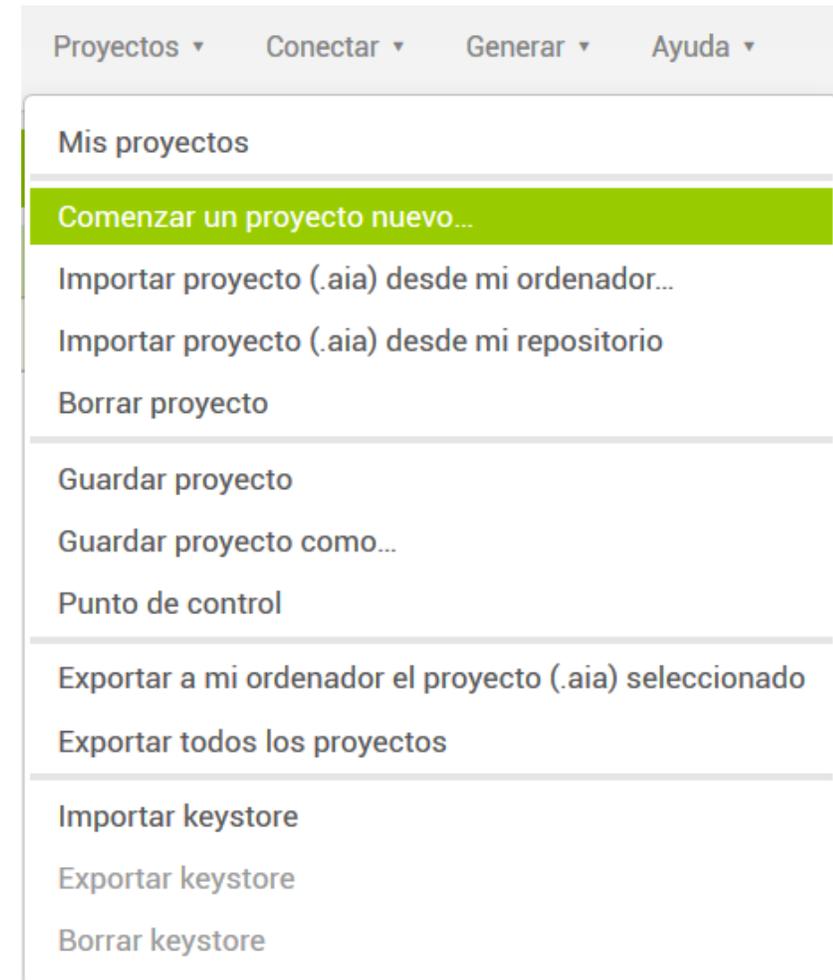


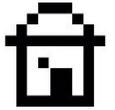
Primera App: sumas

- **Paso 1:** dirígete a la barra de herramientas y en la pestaña **Proyectos** escoge la opción **Comenzar un proyecto nuevo**.
- **Paso 2:** aparecerá una pantalla en la que has de **introducir el nombre** que va a tener la App.

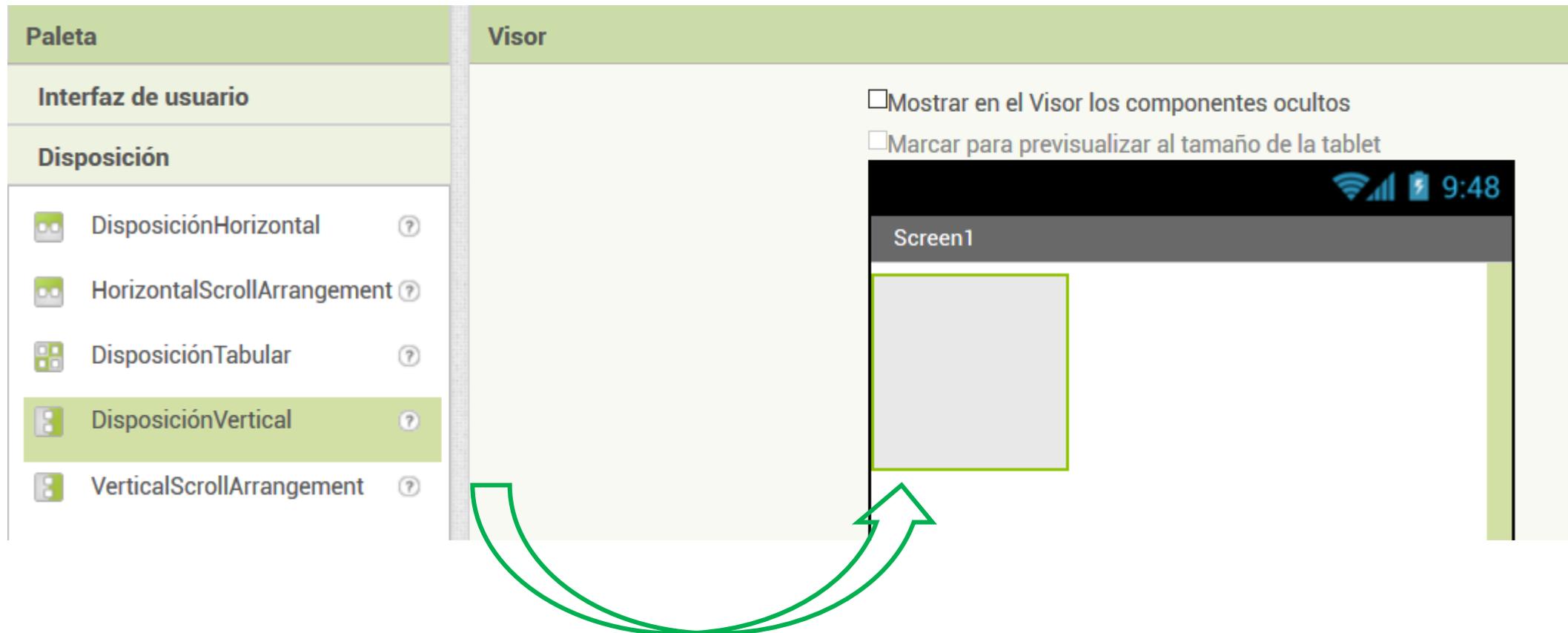
Crear un nuevo proyecto de App Inventor

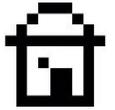
Nombre del proyecto:





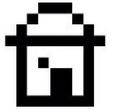
- **Paso 3:** dirígete a la **Paleta** y en la pestaña **Disposición** encontrarás la **DisposiciónVertical**, seguidamente arrástrala al **Visor**.





- **Paso 4:** verás que se ha generado un nuevo objeto en la lista de **Componentes**. A continuación deberás establecer un ancho adecuado para esta capa de disposición desde el apartado **Propiedades**. En el campo llamado Ancho, debes seleccionar la opción **Ajustar al contenedor**, para que así el contenido se ajuste al tamaño de la pantalla del dispositivo.

The screenshot displays the Android Studio interface. On the left, the 'Componentes' panel shows a tree view with 'Screen1' and a sub-component 'DisposiciónVertical1'. On the right, the 'Propiedades' panel is open for 'DisposiciónVertical1'. It shows the following settings: 'DispHorizontal' is set to 'Izquierda : 1', 'DispVertical' is set to 'Arriba : 1', 'ColorDeFondo' is 'Por defecto' (represented by a black square), 'Alto' is 'Automático...', and 'Ancho' is set to 'Ajustar al contenedor' (indicated by a selected radio button). Below the 'Ancho' options are 'Cancelar' and 'Aceptar' buttons.



- **Paso 5:** una vez se haya adaptado el tamaño de la capa a la pantalla del dispositivo, vas a indicar en las propiedades de esta capa que debe situar todo el contenido de su interior de un modo centrado. Para ello ves al apartado **Propiedades** y busca el campo que dice **DispHorizontal** y escoge la opción **Centro**.

Propiedades

Screen1

PantallaAcercaDe

DispHorizontal

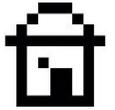
Centro : 3 ▼

DispVertical

Arriba : 1 ▼

AppName

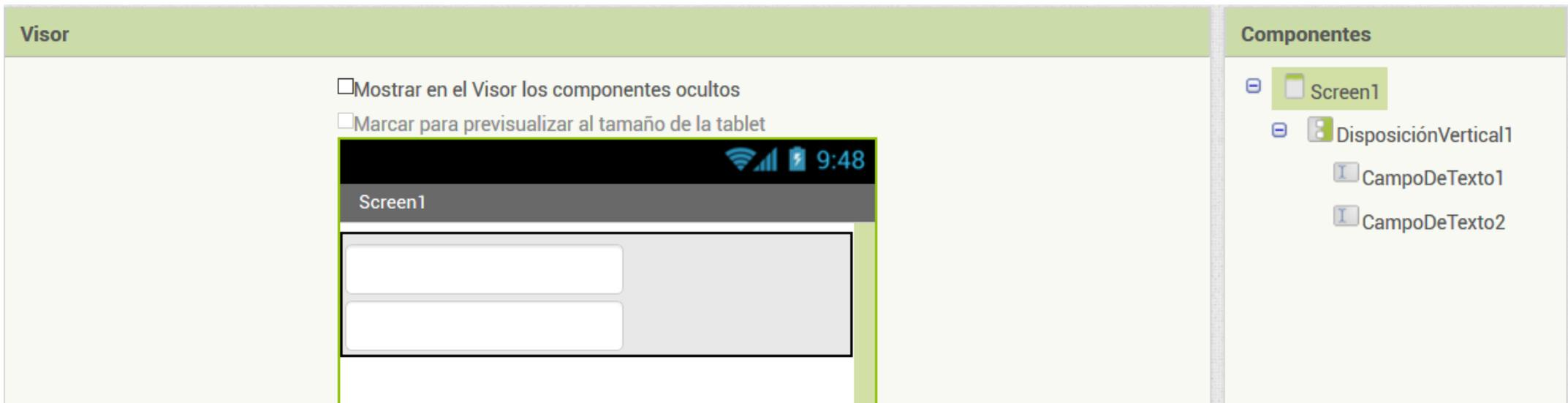
Suma

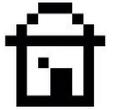


- **Paso 6:** Una vez ya está lista la disposición de los objetos dentro de la App, se van a comenzar a añadir éstos al apartado **Visor**. Lo primero que vas a añadir van a ser los campos donde se introducirán los números que luego serán calculados. Para ello vas a ir a la **Paleta** y en la sección de **Interfaz de usuario** has de arrastrar el elemento **CampoDeTexto** dentro de la capa de disposición agregada anteriormente.

Paleta		
Interfaz de usuario		
	Botón	?
	CasillaDeVerificación	?
	SelectorDeFecha	?
	Imagen	?
	Etiqueta	?
	SelectorDeLista	?
	VisorDeLista	?
	Notificador	?
	CampoDeContraseña	?
	Deslizador	?
	Desplegable	?
	CampoDeTexto	?
	SelectorDeHora	?
	VisorWeb	?

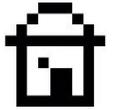
- **Paso 7:** repite los pasos para agregar otro campo de texto debajo del que ya está en la App. Una vez hayas agregado el segundo campo de texto, deberás tener algo como lo que se muestra en la imagen.



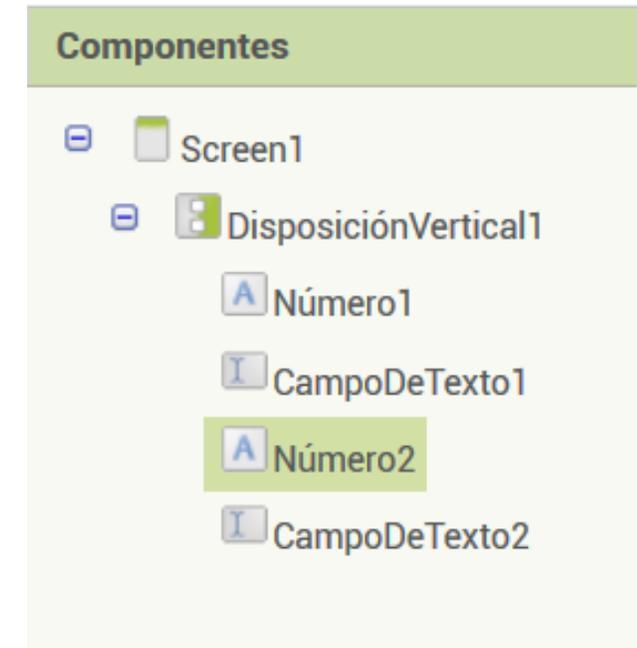


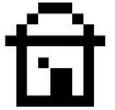
- **Paso 8:** de la misma manera que se han agregado campos de texto, vas a repetir el procedimiento pero para agregar una **etiqueta** que hará de cabecera en los campos agregados anteriormente. Para ello vas a ir a la **Paleta** y en la sección de **Interfaz de usuario** has de arrastrar el elemento **Etiqueta** encima de cada uno de los dos campos de texto.

The screenshot displays the Xamarin.Forms design environment. On the left, the 'Paleta' (Palette) is open to the 'Interfaz de usuario' (User Interface) section, where the 'Etiqueta' (Label) component is selected. The central 'Visor' (Viewer) shows a preview of a mobile screen with a status bar at the top showing the time 9:48. Below the status bar, the screen is titled 'Screen1' and contains two text input fields. Each field has a label above it: 'Texto para Etiqueta1' and 'Texto para Etiqueta2'. The 'Etiqueta2' label is highlighted with a green border. On the right, the 'Componentes' (Components) panel shows a hierarchical tree of the UI elements: 'Screen1' contains 'DisposiciónVertical1', which contains 'Etiqueta1', 'CampoDeTexto1', 'Etiqueta2', and 'CampoDeTexto2'. The 'Etiqueta2' component is highlighted in green.

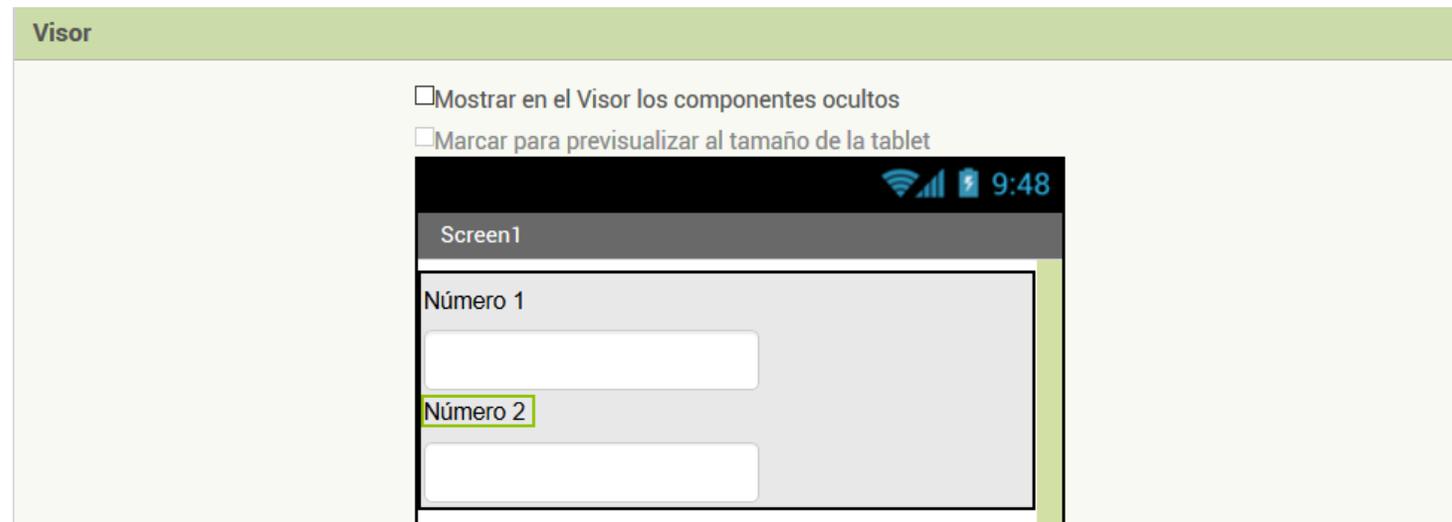


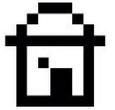
- **Paso 9:** A continuación vas a renombrar estos componentes para hacer mas fácil el trabajo con ellos. Para ello selecciona el componente deseado y haz clic en el botón **Cambiar nombre**. Poniendo Número 1 y Número 2. Fíjate que aunque hasta cambiado el nombre de los componentes, en el visor continúa apareciendo “Texto para etiqueta 1” y “Texto para etiqueta 2”. Vamos a ver cómo cambiar esto.



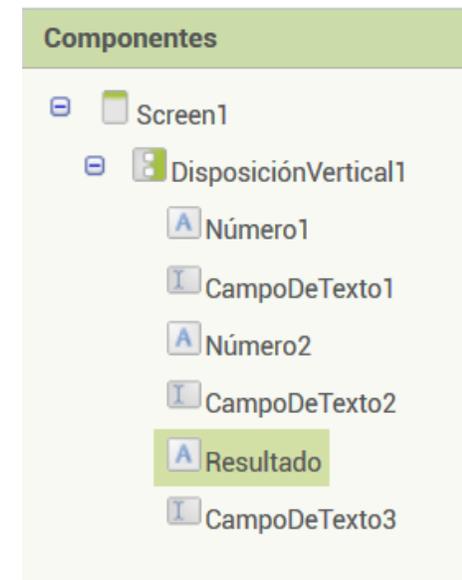


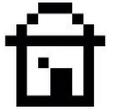
- **Paso 10:** ahora que ya tienes los componentes renombrados para localizarlos mas fácilmente, vas a modificar los textos que muestran por pantalla estos mismos componentes. Solamente va a ser necesario que modifiques el texto de las etiquetas que hacen de cabecera. Para ello marca la etiqueta correspondiente en el visor, ves a al apartado de **Propiedades** y modifica el campo **Texto**. Te debería quedar algo así.





- **Paso 11:** del mismo modo que has agregado campos de texto y etiquetas para introducir los números a calcular, vas a tener que replicar los pasos para agregar otro campo de texto y otra etiqueta, los cuales servirán para mostrar el resultado del cálculo a realizar. Deberá quedar algo como lo que se muestra en la imagen.

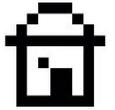




Primera App: sumas

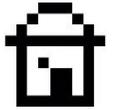
- **Paso 12:** Ahora ya tienes preparada la App para introducir los números a calcular y para que muestre el resultado. A continuación vas a agregar un **Botón** para que estos componentes sepan en qué momento realizar las operaciones. Para ello ves al apartado de la **Paleta** y en la sección de **Interfaz de usuario** has de arrastrar el elemento **Botón** debajo del campo resultado.





- **Paso 13:** ya está finalizada toda la parte de diseño, por lo que ahora deberás agregar **bloques (instrucciones)** a los componentes desde el apartado **Bloques** de la interfaz. Para llevar a cabo esta tarea lo primero es plantear como va a funcionar la app:
El botón Suma va a recoger los valores de los campos 1 y 2, sumarlos y mostrar el resultado en el campo 3.
- **Paso 14:** en el apartado **bloques**, primero selecciona el componente del botón “Suma” para que se muestren todos los bloques disponibles para este tipo de objeto. En este caso vamos a querer que este botón inicie una serie de instrucciones. Busca el siguiente bloque **“Cuando Sumar .Clic”**.

Primera App: sumas



The image shows the Android Studio interface for a simple addition application. The left pane, titled "Bloques", displays a hierarchy of UI components under "Integrados". The right pane, titled "Visor", shows the visual programming logic for the "Suma" button.

Bloques (Left Pane):

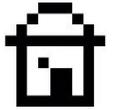
- Integrados
 - Control
 - Lógica
 - Matemáticas
 - Texto
 - Listas
 - Colores
 - Variables
 - Procedimientos
- Screen1
 - DisposiciónVertical1
 - Número1
 - CampoDeTexto1
 - Número2
 - CampoDeTexto2
 - Resultado
 - CampoDeTexto3
 - Suma

Visor (Right Pane):

The visual programming logic for the "Suma" button consists of six event listener blocks:

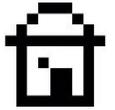
- cuando Suma .Clic ejecutar
- cuando Suma .ObtenertFoco ejecutar
- cuando Suma .ClicLargo ejecutar
- cuando Suma .PerderFoco ejecutar
- cuando Suma .Presionar ejecutar
- cuando Suma .Soltar ejecutar

At the bottom of the right pane, there is a block for "Suma . ColorDeFondo", which is currently selected.



- **Paso 15:** ahora ya está colocado el bloque que reaccionará cuando pulses el botón “Sumar” de la App, pero todavía hay que decirle qué deseas que haga, en este caso es mostrar en el campo 3 la suma de dos números. Para ello vas a hacer clic encima del **CampoDeTexto3** en bloques y a continuación buscarás el bloque llamado: **Poner campo_3.text como** y lo encajas en el bloque previo.



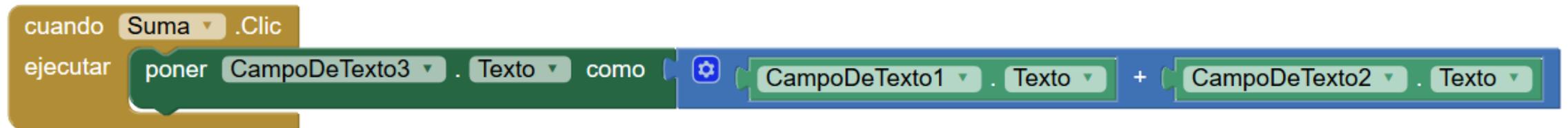


- **Paso 16:** ahora es el momento de indicar al campo 3 de donde va a recoger los números para posteriormente sumarlos y mostrarlos. Como los valores van a recibir un tratamiento matemático, vamos a buscar en la sección **Matemáticas** el bloque que se encarga de **Sumar**, y lo vas a unir con el conjunto de bloques anterior.

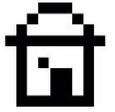




- **Paso 17:** por último falta añadir el valor de los campos 1 y 2 a los huecos que propone el campo matemático. Ya que los datos que incluyen los campos 1 y 2 son representados en texto, vas a hacer clic en cada uno de éstos y buscarás el bloque de **Texto** para arrastrarlos al interior del bloque matemático, para obtener algo así.



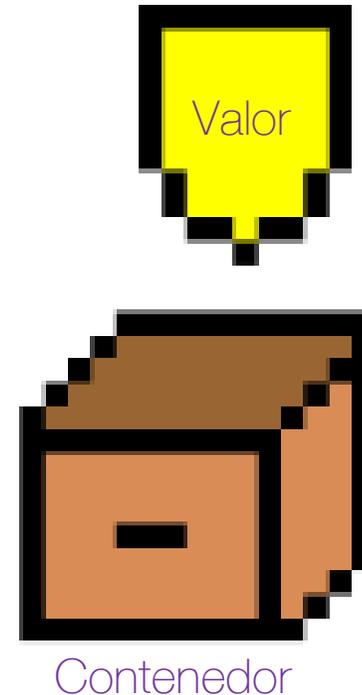
¡Primer programa terminado! Pruébalo con el emulador o pasándolo a tu móvil con AI Companion!

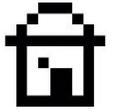


5. SEGUNDA APP: PAR O IMPAR

Como segunda actividad práctica, vas a realizar una App que se encargará de decirte si el número que introduzcas, es del tipo par o impar.

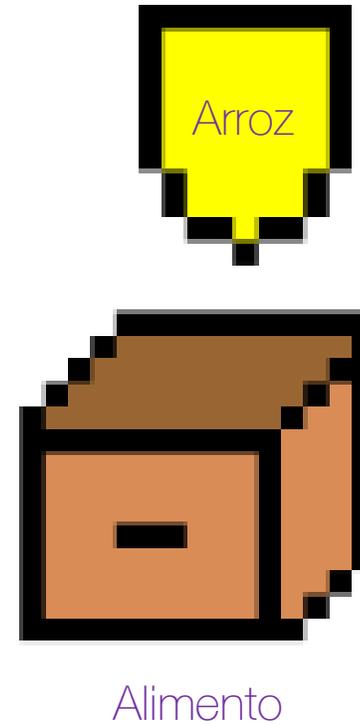
Para realizar esta práctica, primero vas a conocer un nuevo concepto: **las variables**.

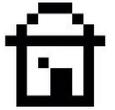




Una variable la puedes imaginar como una caja donde almacenas algo, que luego podrás volver a reutilizar, ya que siempre estará almacenado en un lugar llamado variable con un nombre concreto.

Puedes crear las variables con un nombre que identifiquen el contenido de éstas. Así por ejemplo puedes tener una variable “alimento” que puede contener diferentes valores: “arroz”, “pan”, “fruta”, etc.





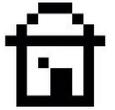
Segunda App: par o impar

- **Paso 1:** dirígete a la barra de herramientas y en la pestaña **Proyectos** escoge la opción **Comenzar un proyecto nuevo**.
- **Paso 2:** aparecerá una pantalla en la que has de introducir el nombre con el que se va a nombrar esta App. Escribimos “Par_impar”.
Recuerda que los nombres de proyecto no pueden tener espacios.

Crear un nuevo proyecto de App Inventor

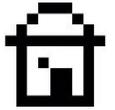
Nombre del proyecto:

Cancelar Aceptar



- **Paso 3:** dirígete a la **Paleta** y en la pestaña **Disposición** encontrarás la **DisposiciónVertical**, seguidamente arrástrala al **Visor**.





Segunda App: par o impar

- **Paso 4:** verás que se ha generado un nuevo objeto en la lista de **Componentes**. A continuación deberás establecer un ancho adecuado para esta capa de disposición desde el apartado **Propiedades**. En el campo llamado **Ancho**, debes seleccionar la opción **Ajustar al contenedor**.

The screenshot shows the Android Studio interface with two panels: 'Componentes' and 'Propiedades'. In the 'Componentes' panel, a tree view shows 'Screen1' containing 'DisposiciónVertical1'. The 'Propiedades' panel is open for 'DisposiciónVertical1' and shows the following settings:

- DispHorizontal:** Izquierda : 1
- DispVertical:** Arriba : 1
- ColorDeFondo:** Por defecto (represented by a black square)
- Alto:** Automático...
- Ancho:** A list of options: Automático, Ajustar al contenedor, [] pixels, [] percent.

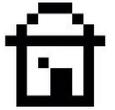
At the bottom of the 'Ancho' section, there are 'Cancelar' and 'Aceptar' buttons.



Segunda App: par o impar

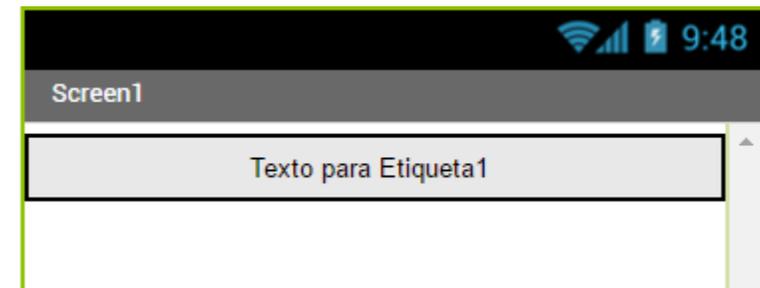
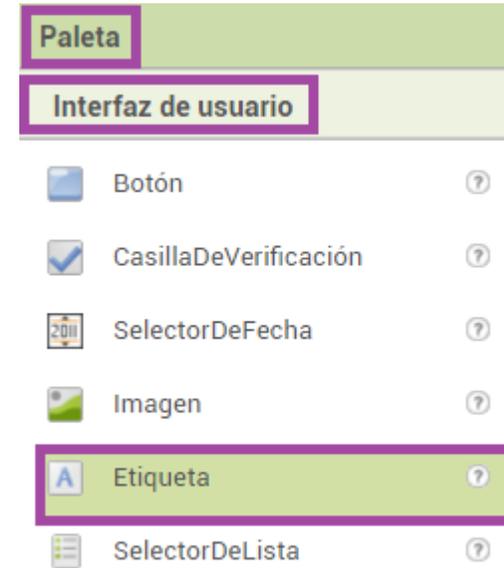
- **Paso 5:** una vez se haya adaptado el tamaño de la capa a la pantalla del dispositivo, vas a indicar a esta capa en sus propiedades que debe situar todo el contenido de su interior de un modo centrado. Para ello ves al apartado **Propiedades** y busca el campo que dice **DispHorizontal** y escoge la opción **Centro**.

The screenshot shows the Visual Studio IDE interface. On the left, the 'Componentes' (Components) pane shows a hierarchy: 'Screen1' containing 'DisposiciónVertical1'. On the right, the 'Propiedades' (Properties) pane is open for 'DisposiciónVertical1'. The 'DispHorizontal' property is expanded, showing a dropdown menu with four options: 'Izquierda : 1', 'Centro : 3' (which is selected and highlighted in green), and 'Derecha : 2'. Below this, other properties are visible: 'Alto' (set to 'Automático...'), 'Ancho' (set to 'Ajustar al contenedor...'), 'Imagen' (set to 'Ninguno...'), and 'Visible' (checked).



Segunda App: par o impar

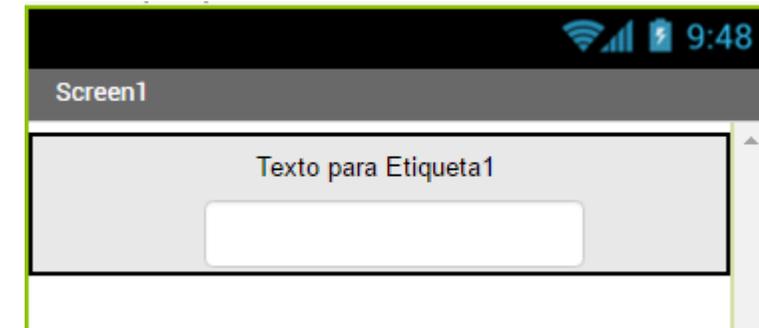
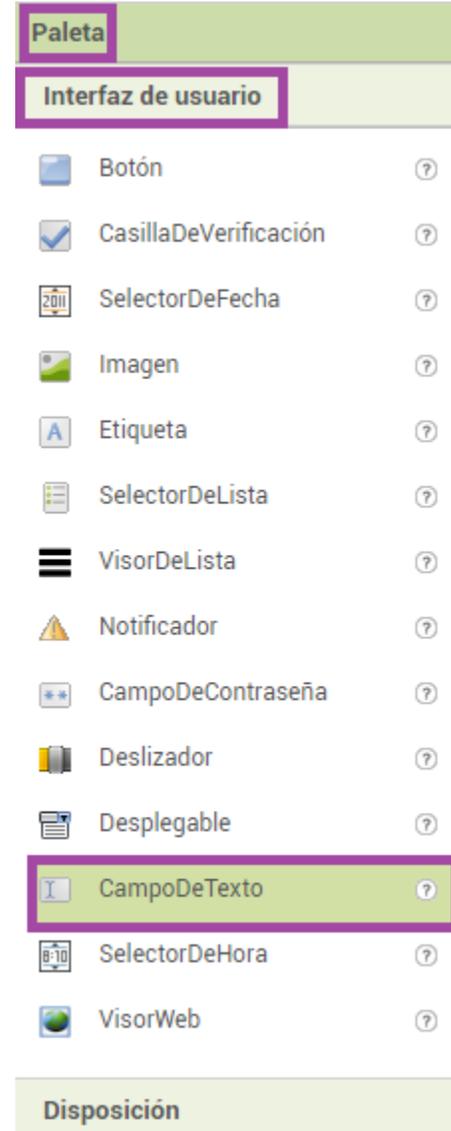
- **Paso 6:** Una vez ya está lista la disposición de los objetos dentro de la app, se van a comenzar a añadir éstos al apartado **Visor**. Lo primero que vas a añadir va a ser un título sobre el campo donde se introducirán los números que luego serán verificados. Para ello vas a ir a la **Paleta** y en la sección de **Interfaz de usuario** has de arrastrar el elemento **Etiqueta** dentro de la capa de disposición agregada anteriormente.

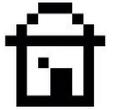




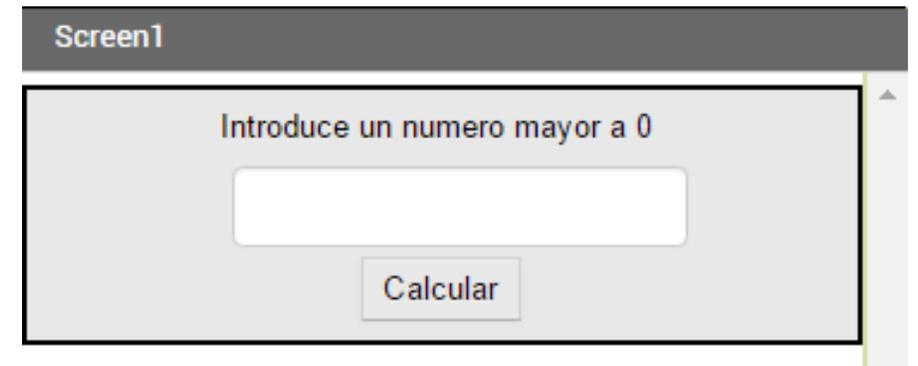
Segunda App: par o impar

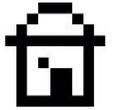
- **Paso 7:** ahora deberás añadir el campo donde se introducirán los números que luego serán calculados. Para ello vas a ir a la **Paleta** y en la sección de **Interfaz de usuario** has de arrastrar el elemento **CampoDeTexto** dentro de la capa de disposición agregada anteriormente.





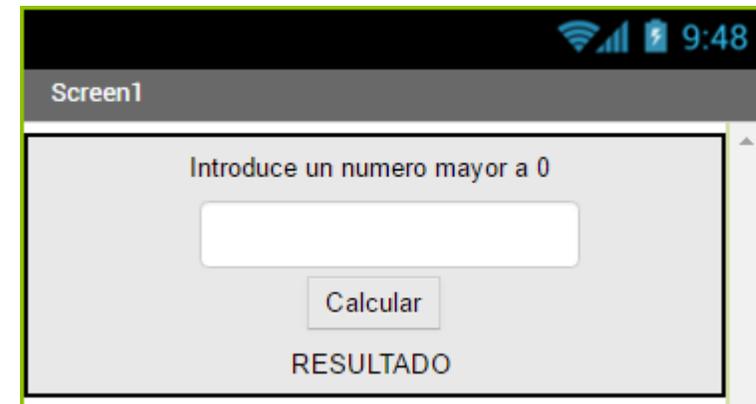
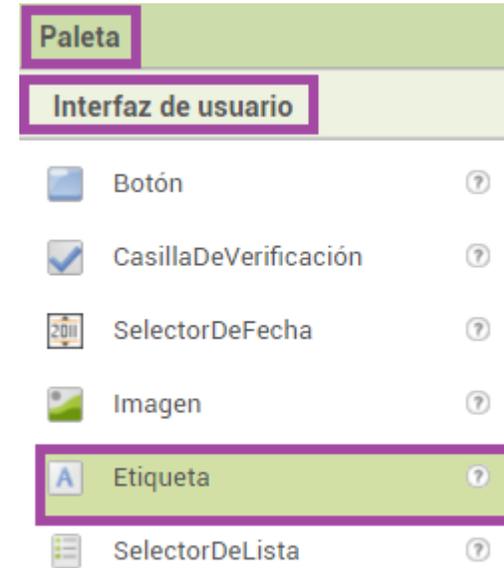
- **Paso 8:** ahora deberás añadir el botón que ejecutará la comparación de los dos números, mediante el uso de instrucciones encapsuladas en bloques. Para ello vas a ir a la **Paleta** y en la sección de **Interfaz de usuario** has de arrastrar el elemento **Botón** dentro de la capa de disposición. Como veis en las imágenes, se han modificado los textos que se muestran por pantalla.

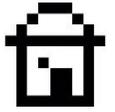




Segunda App: par o impar

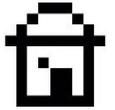
- **Paso 9:** para finalizar la parte visual de la App vas a añadir una etiqueta que mostrará un texto dependiendo del resultado del calculo previo. Deberá mostrar si es Par o Impar el numero añadido. Para ello vas a ir a la **Paleta** y en la sección de **Interfaz de usuario** has de arrastrar el elemento **Etiqueta** dentro de la capa de disposición agregada anteriormente, tal y como se muestra en el **Paso 6**.





- **Paso 10:** ya está preparada la parte visual de la App, pero ahora falta añadir instrucciones a los objetos visuales que se han añadido. Es esencial almacenar datos para posteriormente tratarlos. Para ello vas a crear dos **Variables** para almacenar el número a calcular y el resultado calculado. Vamos a la sección Variables e inicializamos 2 variables.

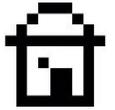




- **Paso 11:** seguidamente vas a hacer clic en el objeto **Botón** que has añadido antes, de manera que van a aparecer las instrucciones asociadas a este bloque. Tal y como se muestra en la imagen vas a agregar al programa el bloque llamado:
Cuando Botón1.Clic



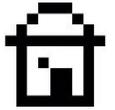
Este bloque ejecutará las instrucciones que contenga en su interior al hacer clic en el objeto Botón1.



- **Paso 12:** ahora vas a indicar que una vez se haga clic en el objeto, se van a coger los datos que haya dentro del **CampoDeTexto1** para almacenarlo en una variable que ya se ha creado antes. Tal y como ves en la imagen, se indica qué valor se debe almacenar, siendo el que tenga la propiedad de **Texto**. Esto se almacena en la variable **numero**.



Ejemplo: El número que se escriba en la **CajaDeTexto** se va a almacenar en la variable llamada **numero**

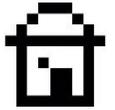


- **Paso 13:** ahora vas a hacer uso de la segunda variable creada con el nombre de **resto**, en la cual vas a almacenar el resultado de calcular el (módulo o resto resultado de dividir por 2) del valor guardado en la variable **numero**. Tal y como ves se añade un modulo matemático (**módulo de**) en el cual introducimos la variable **numero** para dividirla entre 2.



```
cuando Botón1 .Clic
ejecutar
  poner global numero a CampoDeTexto1 .Texto
  poner global resto a módulo de tomar global numero ÷ 2
```

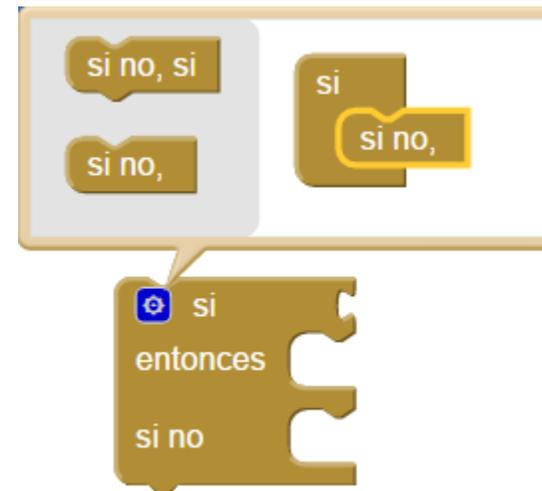
La variable **resto** almacenará un número resultante que servirá como **condicional** en el siguiente bloque a añadir.



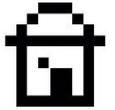
- **Paso 14:** utilizando los bloques de control, en este caso del tipo **si entonces** podrás controlar mediante condiciones, qué instrucciones vas a ejecutar y cuales no. En este caso, si la condición da como respuesta **falso** querrás que se ejecuten las acciones que substituyen a las de una condición **cierta**. Tal y como ves en la imagen puedes modificar este tipo de bloques.



Arrastra el bloque por defecto y haz clic en la rueda azul para modificar el bloque.



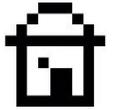
Añade un **si no,** para hacer que si es **FALSA** la condición se ejecuten otros bloques.



- **Paso 15:** tal y como has visto, vas a tener que añadir un bloque del tipo **(si, si no)**, con una condición anidada dentro para que dependiendo del resultado matemático, se ejecute una acción u otra. Como condición vas a tener que decir que si la variable llamada **resto** da como resultado 0 significa que el modulo calculado corresponde a un numero **par**, en caso que no sea 0 será un numero **impar**.



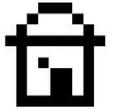
Depende de esta condición que se ejecuten unas u otras acciones.



- **Paso 16:** por último vas a tener que especificar qué acciones se van a ejecutar en cada uno de los dos casos:
 - Si es **cierto** se va a poner como texto en la **Etiqueta2** “número par”,
 - Si es **falso** será el texto “número impar”.

```
cuando Botón1 .Clic
ejecutar
  poner global numero a CampoDeTexto1 . Texto
  poner global resto a módulo de tomar global numero ÷ 2
  si
    tomar global resto = 0
  entonces
    poner Etiqueta2 . Texto como " Numero PAR "
  si no
    poner Etiqueta2 . Texto como " Numero IMPAR "
```

Estas serían las acciones a poder realizar, dependiendo del resultado de la condición.

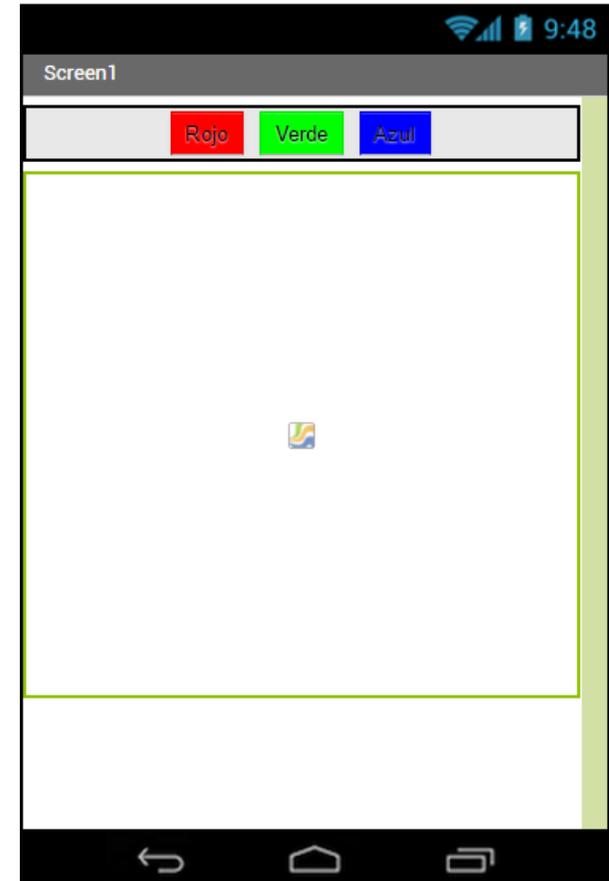


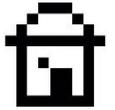
6. TERCERA APP: LIENZO

En esta actividad vas a aprender a dibujar dentro de tus App mediante un componente llamado **Lienzo**.

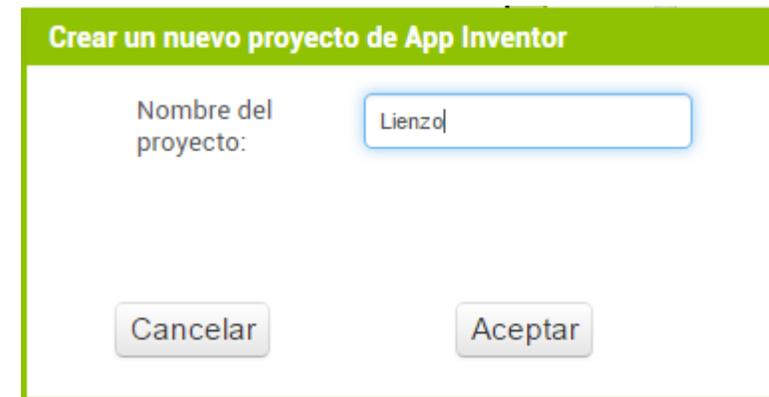
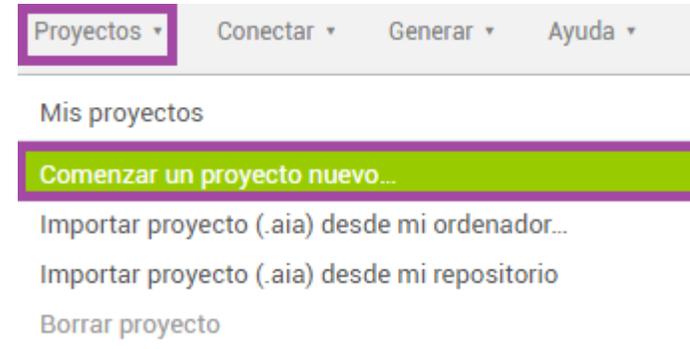
Esta aplicación va a constar de tres botones, que se encargarán de definir el color con el que se va a dibujar dentro del Lienzo.

¿Empezamos?



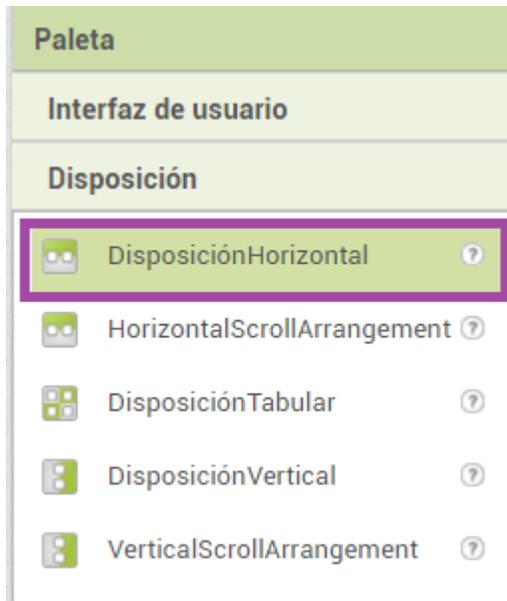


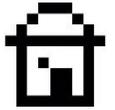
- **Paso 1:** Dirígete a la barra de herramientas y en la pestaña **Proyectos** escoge la opción **Comenzar un proyecto nuevo**.
- **Paso 2:** Aparecerá una pantalla en la que has de introducir el nombre para esta App: lienzo.



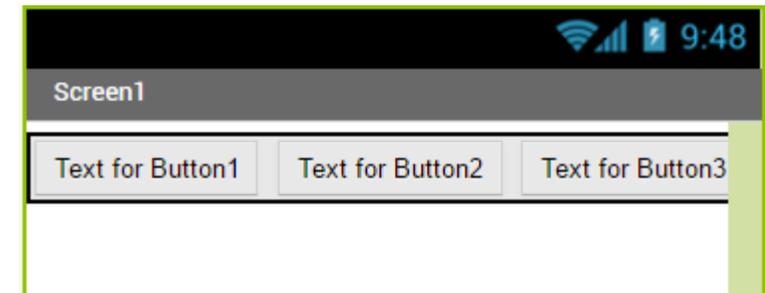
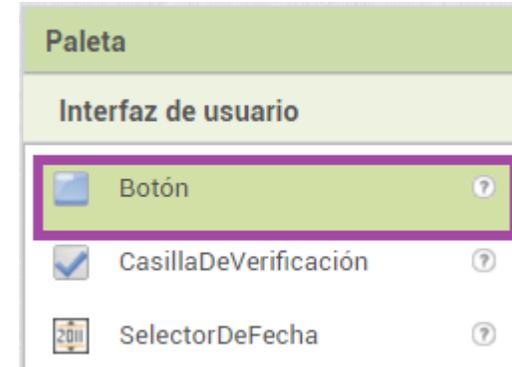


- **Paso 3:** dirígete a la **Paleta** y en la pestaña **Disposición** encontrarás la **DisposiciónHorizontal**, seguidamente arrástrala al **Visor**.

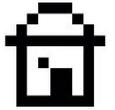




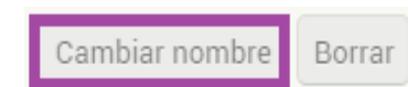
- **Paso 4:** ahora deberás agregar los botones que van a establecer el color del dibujo. Para ello debes dirigirte a la **Paleta** y en la pestaña **Interfaz De Usuario** encontrarás el componente **Botón**. Seguidamente arrástralo al **Visor**.



Para agregar más opciones de colores, en vez de un único botón, agregarás tres en total.



- **Paso 5:** una vez hayas agregado los botones dentro de la capa de distribución, deberías tener en la lista de componentes algo como lo que ves en la imagen de la derecha.
- **Paso 6:** una vez agregados los botones, deberás renombrarlos para poder identificarlos de manera mas fácil.

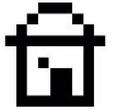




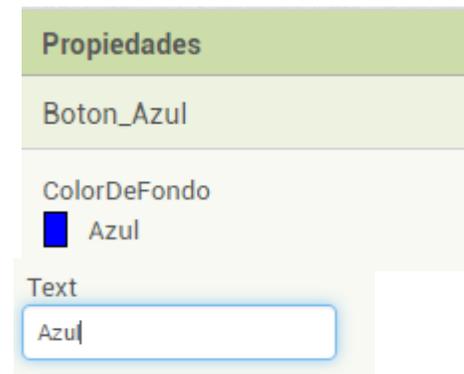
- **Paso 7:** deberás tener algo similar a lo que se muestra en la imagen de la derecha, una vez renombres los tres botones que hacen referencia al color que se va a utilizar a la hora de dibujar en el lienzo.

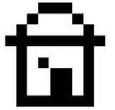


Recuerda que el nombre que le acabas de dar a los botones, no es el texto que se muestra por el Visor cuando la aplicación está iniciada.

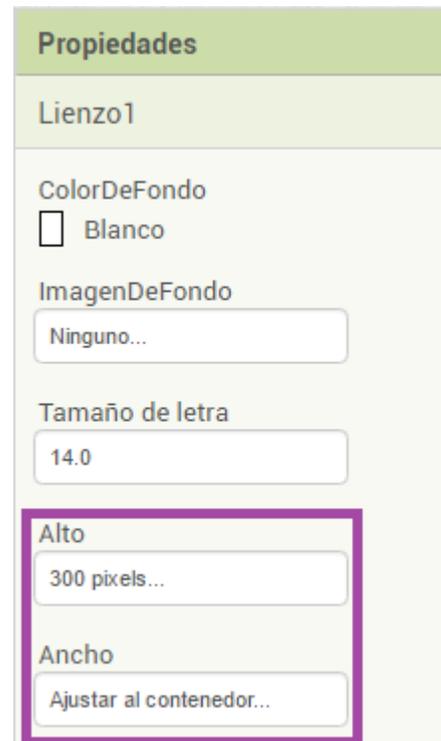


- **Paso 8:** ahora sí es el momento de cambiar tanto el texto que se muestra, como el aspecto visual de los botones. Mediante las **propiedades** de cada uno de los objetos, vas a modificar el color de fondo y el texto, tal y como se muestra en las imágenes.





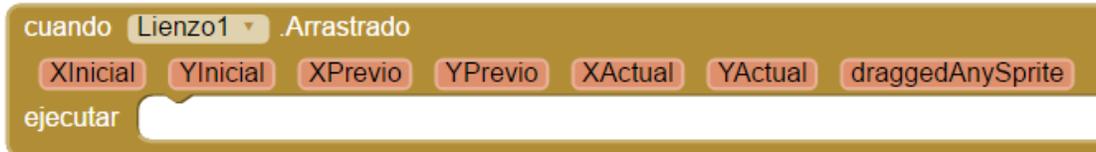
- **Paso 9:** seguidamente vas a agregar el componente **lienzo**, desde la **Paleta** en el apartado de **Dibujo y animación**. Arrástralo al **Visor** y establece sus dimensiones desde las **Propiedades** del Lienzo.



Ancho: Ajustar al contenedor
Alto: 300 Pixel



- **Paso 10:** debes indicarle al componente **lienzo**, qué tiene que hacer cuando se arrastra el dedo por encima. En este caso vas a decirle que genere una línea a medida que vas arrastrando el dedo por la pantalla. Para ello selecciona el componente **lienzo** y busca el bloque que se muestra en la imagen.



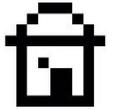
Como puedes ver en la imagen del bloque, éste contiene una serie de parámetros predefinidos que aportan la información necesaria para que en este caso se dibuje una línea. En este ejercicio vas a tener que utilizar los siguientes parámetros: **Xprevio**, **Yprevio**, **Xactual** e **Yactual**.



- **Paso 11:** a continuación, dentro del bloque, debes escoger la rutina predefinida llamada **.DibujarLínea**

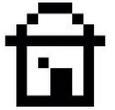


Una vez está agregada dentro del bloque, vas a añadir los parámetros tal y como ves en la segunda imagen.



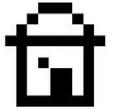
- **Paso 12:** por último, vas a decirle a los botones que has agregado anteriormente, que establezcan el color que marca el texto, es decir, al botón Verde le dirás que establezca dicho color para que dibujes. De este mismo modo repetirás el proceso con los tres botones. Para ellos haz clic en el componente de uno de los botones, y agrega el bloque **Cuando *Botón* .Clic**





- **Paso 13:** una vez agregado este bloque para cada botón, vas a seleccionar el componente **lienzo** para decirle a los botones el color de la línea específico para cada uno. Tal y como ves en la imagen, deberás tener algo así al finalizar el programa.

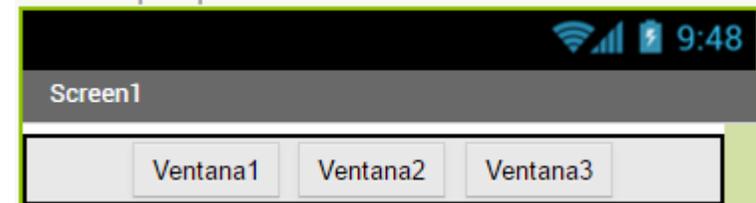


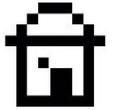


7. CUARTA APP: APP MULTIVENTANA

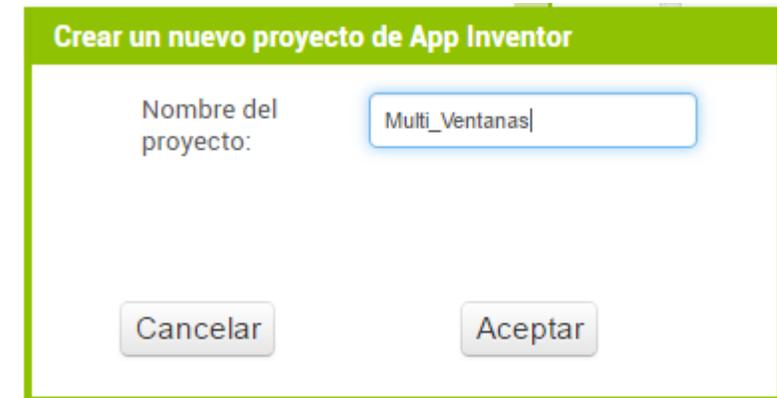
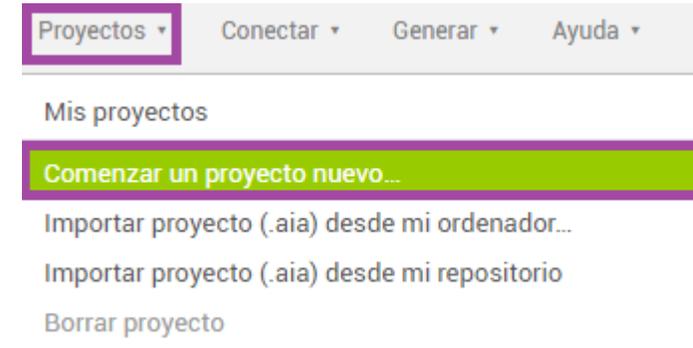
En esta actividad vas a aprender a crear una App con más de una ventana. Para ello deberás agregar más ventanas de navegación y botones para ir abriendo y cerrándolas.

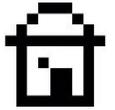
En la imagen de la derecha ves 3 botones, uno para activar cada ventana.



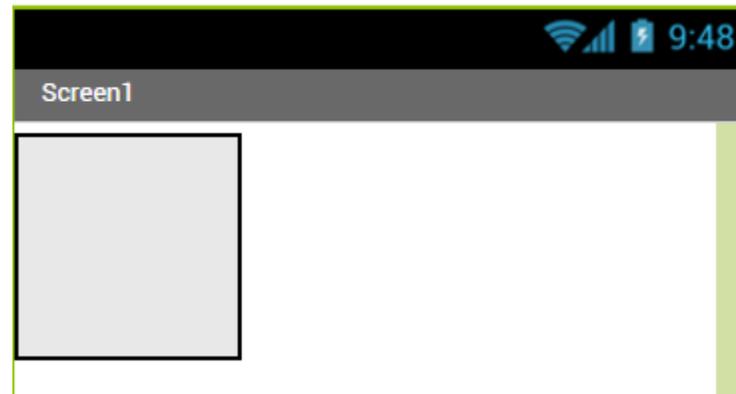
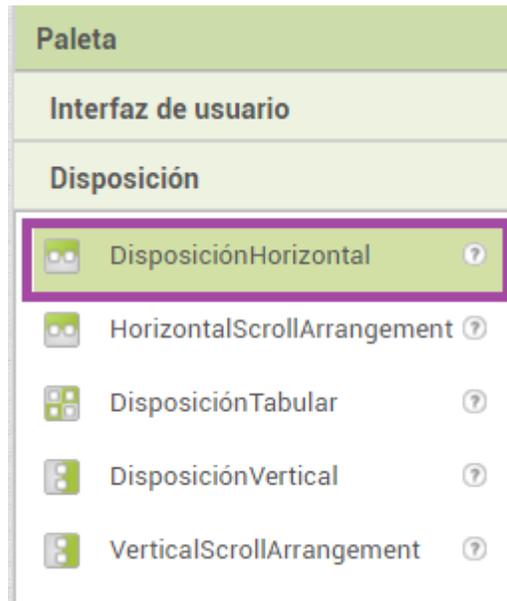


- **Paso 1:** Lo primero que debes hacer, como en todas las App, es dirigirte a la pestaña Proyectos, y crear uno nuevo.
- **Paso 2:** Seguidamente dale un nombre a la App tal y como ves en la imagen.





- **Paso 3:** ahora, como en la gran mayoría de aplicaciones, vas a agregar una capa de distribución para los objetos que se añadirán a continuación. En este caso se va a seleccionar una capa **DisposiciónHorizontal** tal y como ves en la imagen.

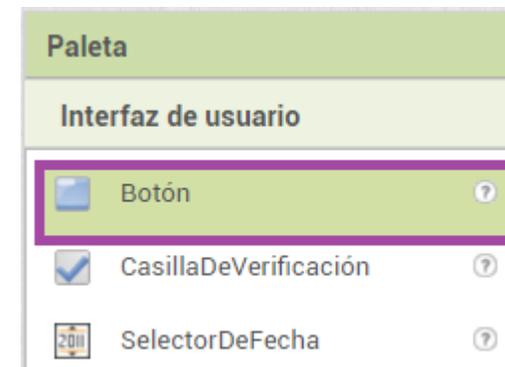
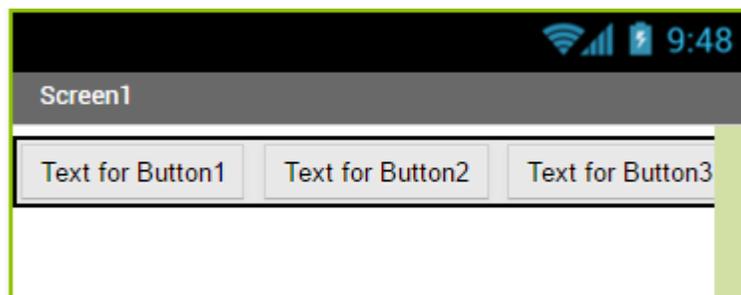


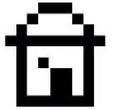


- **Paso 4:** recuerda modificar las propiedades de la capa agregada, de manera que el contenido quede centrado y el ancho de la capa se ajuste al tamaño máximo de la pantalla.

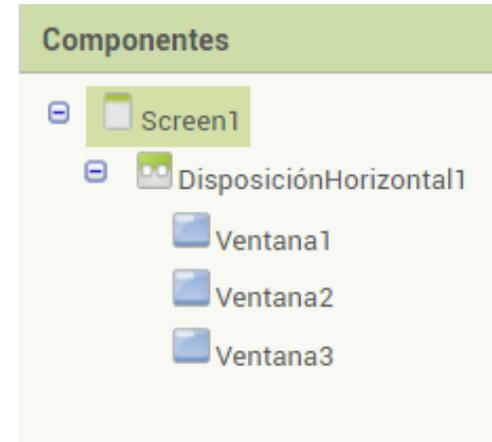
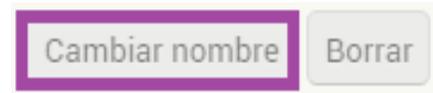


- **Paso 5:** A continuación vas a agregar tres botones que encargarán de llevarte a la ventana deseada cuando haces clic encima.

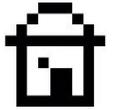




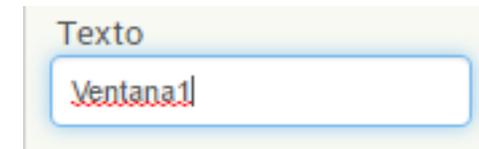
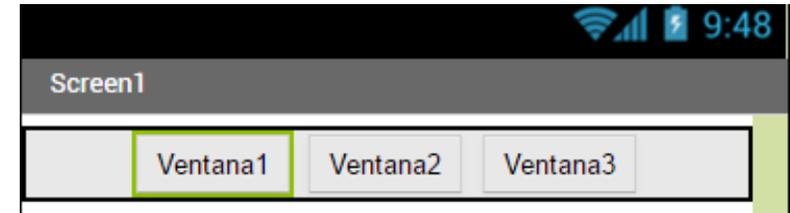
- **Paso 6:** seguidamente vas a modificar el nombre de los botones dentro de la lista de componentes.



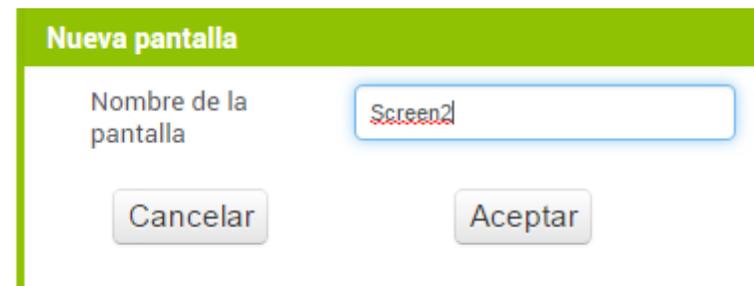
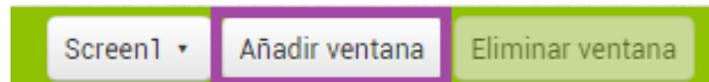
Recuerda que estos nombres son los que después identificarán a los objetos a la hora de ser programados.

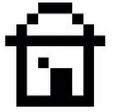


- **Paso 7:** Ahora vas a modificar el texto que muestran los botones por pantalla mediante las propiedades de los componentes tal y como se muestra en la imagen.



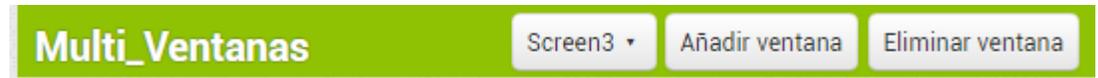
- **Paso 8:** Ahora es el momento de añadir más ventanas para navegar por nuestra App. Tal y como ves en la imagen, haz clic en el botón y añade dos ventanas más, aparte de la que se crea al iniciar un proyecto.

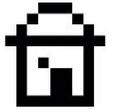




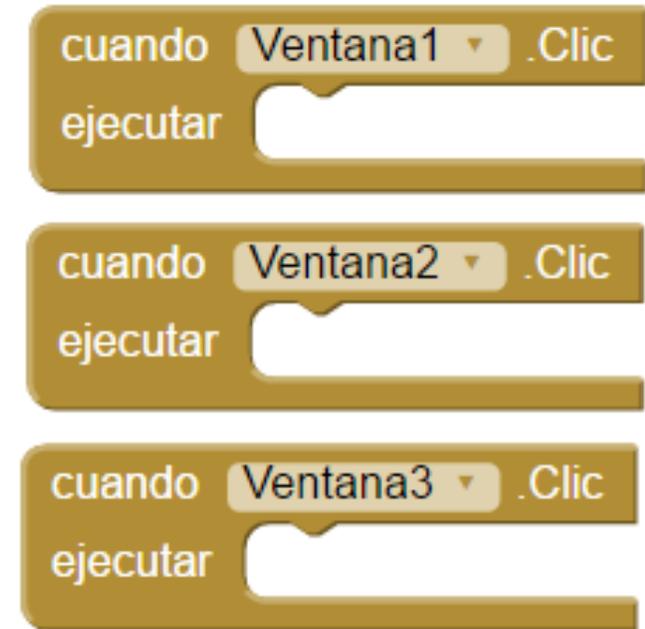
Cuarta App: app multiventana

- **Paso 9:** Una vez tienes todas las ventanas creadas, vas a agregar un botón para ir hacia atrás en la App. Este botón lo pondrás en las ventanas 2 y 3, ya que la primera tendrá los tres botones agregados antes siendo el menú principal de la App. Tal y como ves en la imagen deberás renombrar el botón de cada ventana y ya estará preparada la App para ser programada.





- **Paso 10:** seguidamente vas a comenzar con la programación de los objetos. En primer lugar vas a situarte en la primera ventana **Screen1** y vas a decirle a los objetos botón que reaccionen cuando se hace clic encima de ellos.



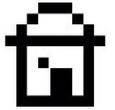
Recuerda que los nombres que ves en las imágenes, son los nombres que le has dado a los botones que van a abrir la ventana escogida.



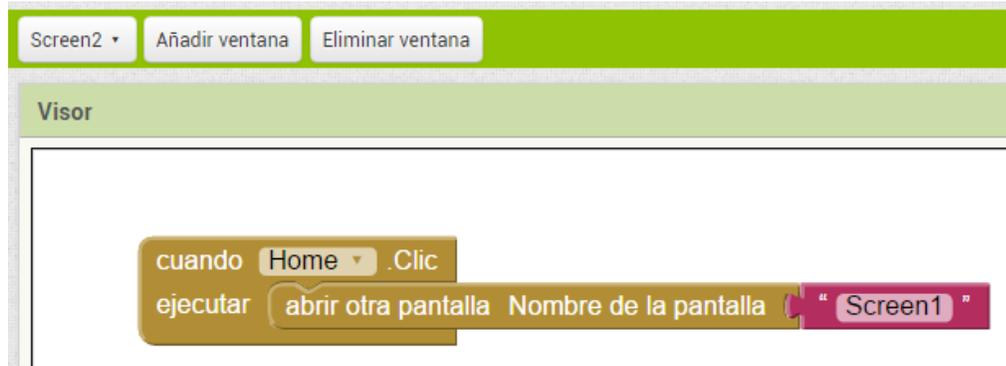
- **Paso 11:** ahora al hacer clic en cualquiera de estos tres botones, vas a indicar que debe abrir la ventana deseada, y para ello vas a seleccionar el conjunto de bloques **Control** y vas a arrastrar dentro de los tres bloques anteriores, el bloque llamado **“open another screen screenName”** (abrir otra pantalla Nombre de la pantalla).

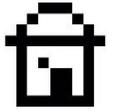


Añade un bloque de texto, con el nombre de la ventana que se desea abrir con cada botón.



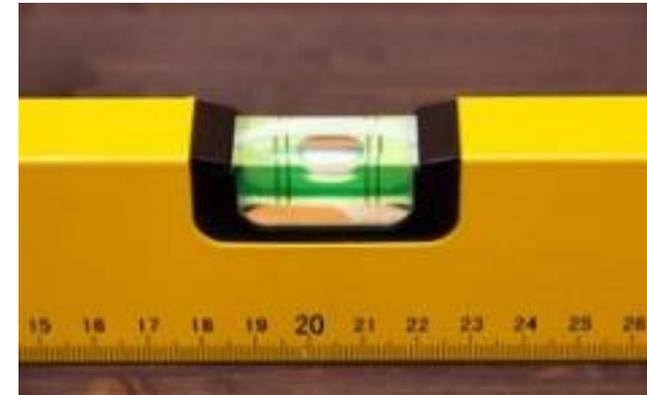
- **Paso 12:** por último repite este mismo proceso en las ventanas 2 y 3 con el botón que hace retroceder hacia la ventana1 (botón de retroceso).

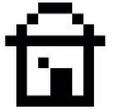




8. QUINTA APP: NIVEL DE BURBUJA

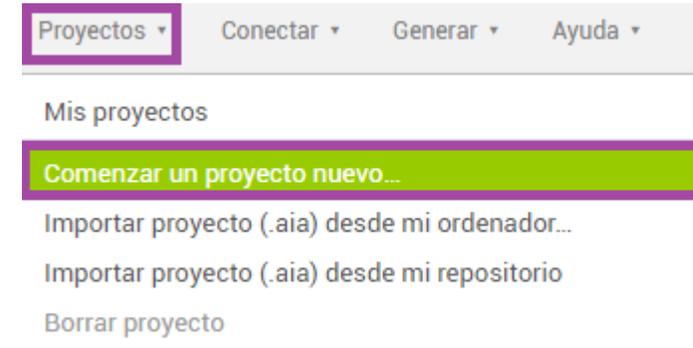
En esta actividad vas a aprender a crear una App la cual podrás utilizar en tu día a día y que aprovecha el **acelerómetro** de los smartphones. Un nivel de burbuja es una herramienta de medición que nos permite comprobar la horizontalidad o verticalidad de un elemento, y que es muy útil por ejemplo, si queremos colgar un cuadro.

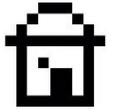




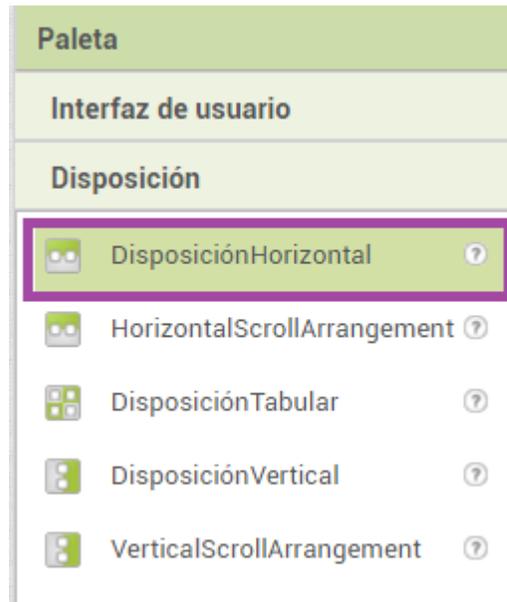
Quinta App: nivel de burbuja

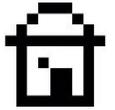
- **Paso 1:** Lo primero que debes hacer, como en todas las Apps, es dirigirte a la pestaña **Proyectos**, y crear uno nuevo.
- **Paso 2:** seguidamente dale un nombre a la App tal y como ves en la imagen.



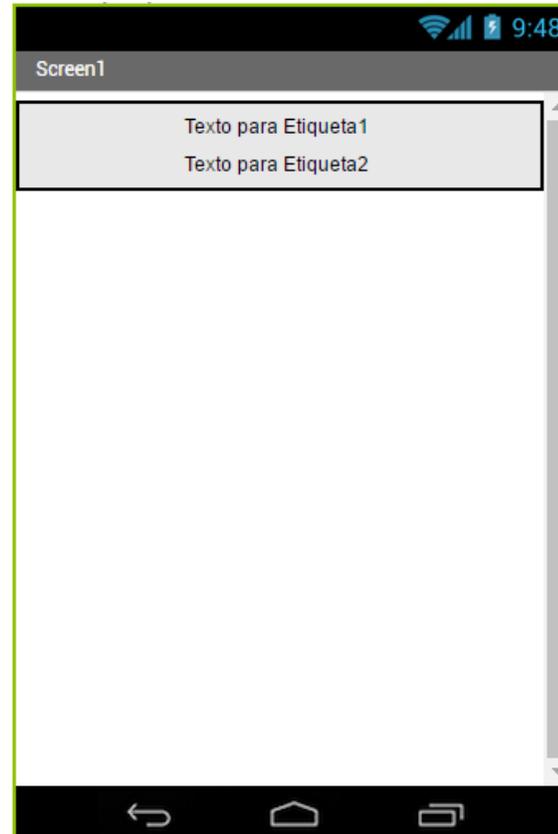
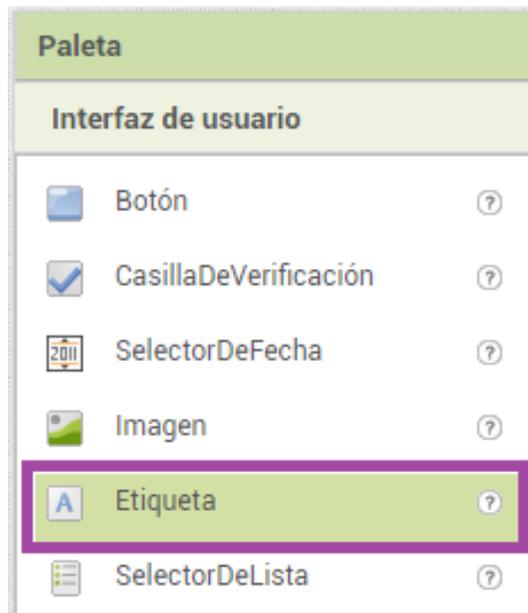


- **Paso 3:** ahora como en casi todas las apps, vas a agregar una capa de distribución para los objetos que se añadirán a continuación. En este caso se va a seleccionar una capa **DisposiciónHorizontal** tal y como ves en la imagen.





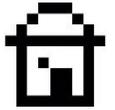
- **Paso 4:** En este ejercicio se van a mostrar datos por la pantalla, de manera que va a agregar **dos etiquetas** dentro de la capa de distribución.



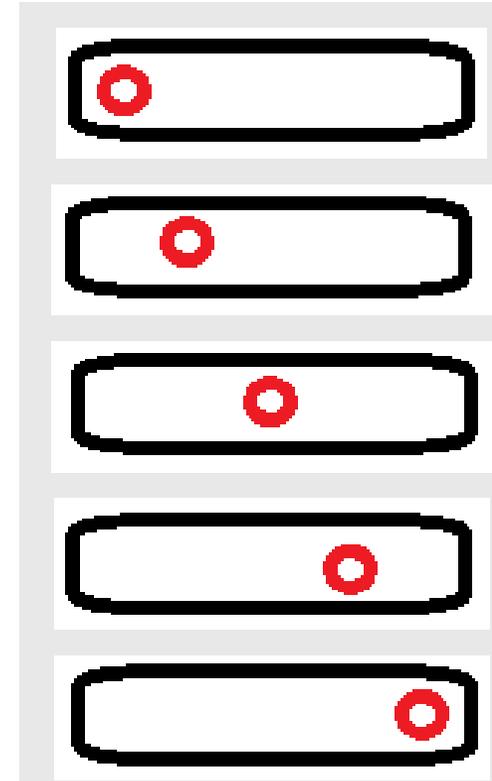


- **Paso 5:** seguidamente vas a ponerle nombre a las etiquetas para localizarlas mejor. Además deberás dejar en blanco el texto que muestran por pantalla, ya que se va a generar de manera automática mediante los bloques de programación que trabajaremos luego.



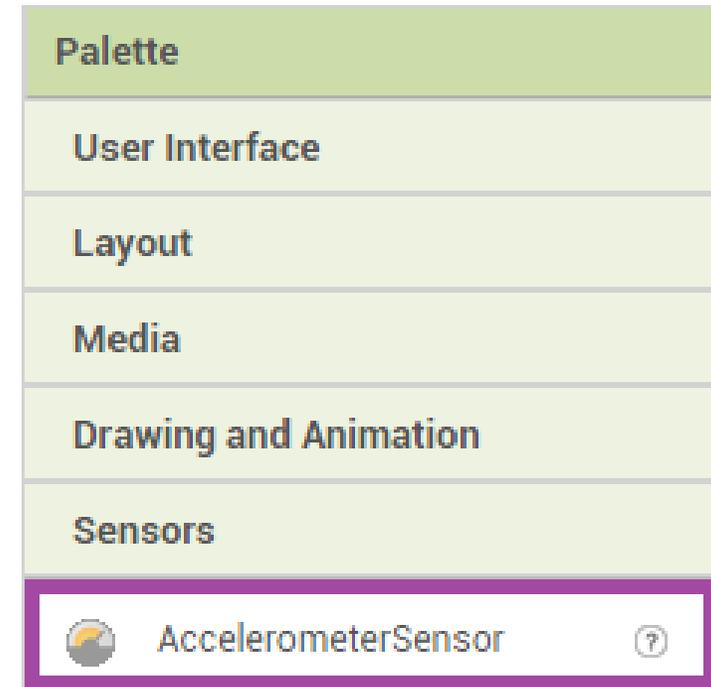


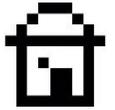
- **Paso 6:** ahora vas a tener que agregar unas **cinco imágenes** como las que ves a la derecha, de manera que se va a crear una animación a tiempo real. Dependiendo de la información que nos proporcione el sensor del dispositivo móvil se va a mostrar una imagen u otra. Esta acción se hace desde el botón **“Subir archivo”**, y deberás realizarla para imagen por separado.





- **Paso 7:** una vez hayas agregado las imágenes de una en una, ya podrás continuar agregando los componentes de sensor desde la pestaña de **Sensores** alojada en la **Paleta** de la interfaz. Este componente se llama **Acelerómetro** y se incluirá en el visor, aunque no estará visible.

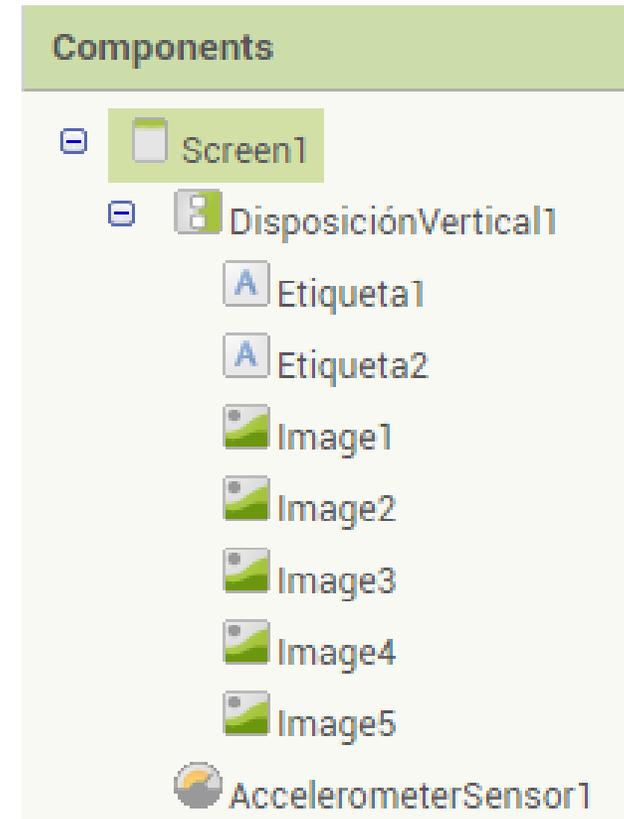


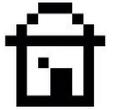


Quinta App: nivel de burbuja

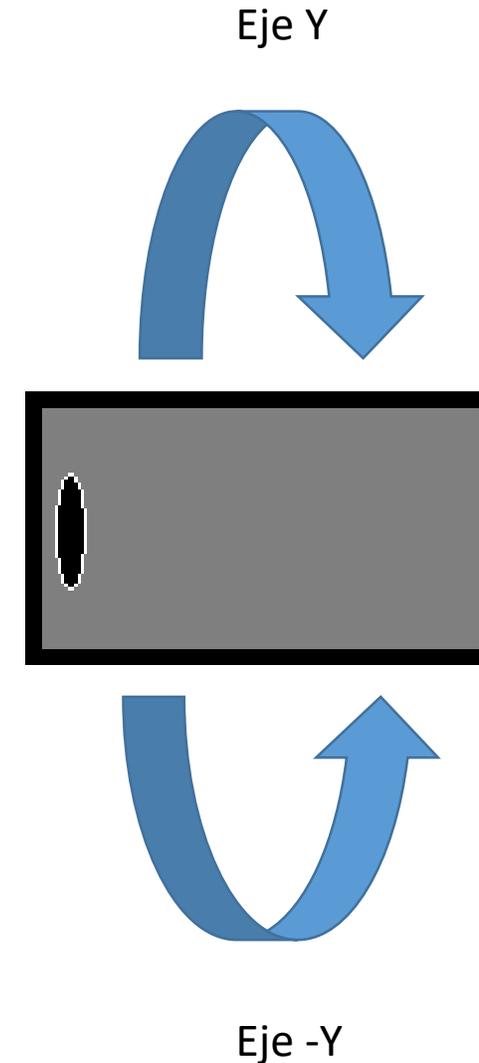
- **Paso 8:** Una vez se hayan realizado los pasos anteriores deberías tener en la lista de componentes algo como lo que se muestra en la imagen. Seguidamente vas a comenzar a agregar bloques para ir programando la aplicación.

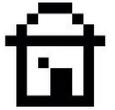
Recuerda poner todas las imágenes como **No visibles**, desmarcando la opción correspondiente en sus **Propiedades**.





- **Paso 9:** comprendamos cómo funcionará esta aplicación que se va a basar en la información que le proporcionará el sensor de aceleración que incorporan los móviles. Dependiendo del grado de inclinación que el sensor capte, se va a mostrar una imagen u otra, generando una animación en la pantalla mediante las imágenes que hemos importado.

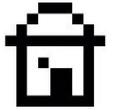




- **Paso 10:** dependiendo de qué tipo de movimiento se realice, el sensor lo traducirá en números negativos o positivos, de manera que gracias a esos números podrás decidir que acciones tomar dentro del programa. Para captar estos cambios de posición, requieres el bloque que se muestra en la imagen. Para localizarlo, haz clic encima del componente **Acelerómetro** y escoge **.CambioEnAceleración**



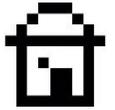
Como ves se pueden captar cambios de los ejes X Y Z



- **Paso 11:** en este caso solo vas a tratar con el **EjeY** ya que es el que te interesa para poder crear una animación en **formato horizontal**. Para tener controlado en todo momento que valor capta el sensor, vas a decirle al programa que cada vez que haya un cambio, lo almacene en una etiqueta que nos va a servir en todo momento.

```
cuando AccelerometerSensor1 .CambioEnAceleración
  xAccel yAccel zAccel
ejecutar poner Etiqueta1 . Texto como
```

Debe quedar de la siguiente manera, con la **Etiqueta1** preparada para recibir los datos cada vez que el sensor capte cambios.



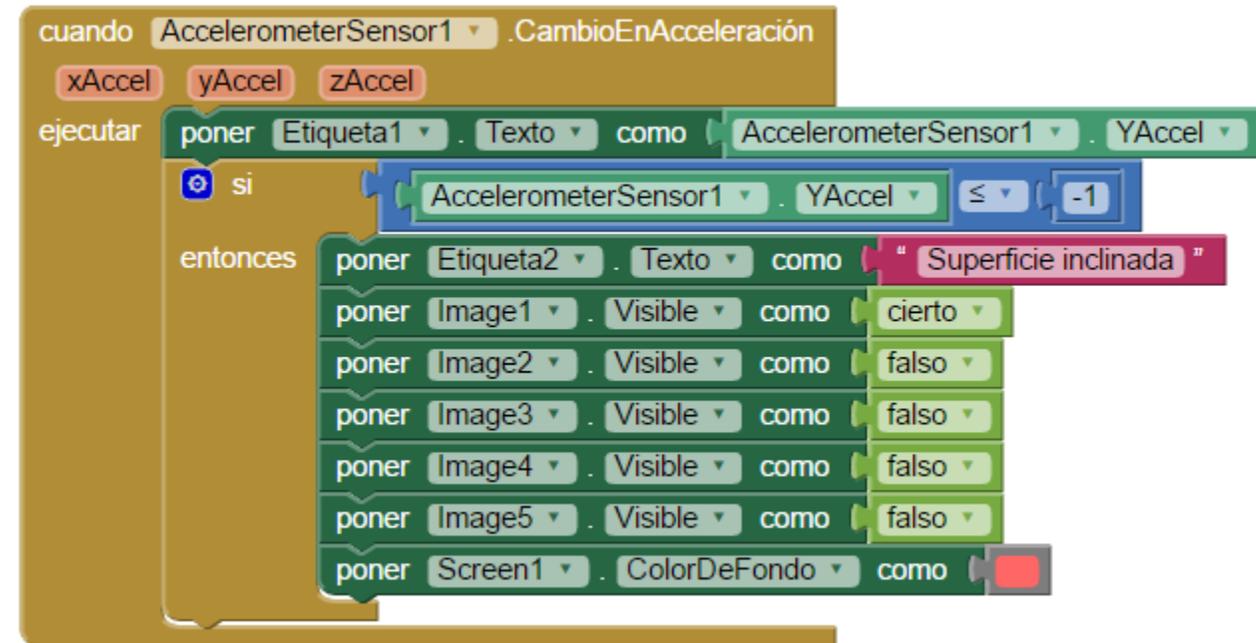
- **Paso 12:** ahora, como ya se ha mencionado, lo que interesa es ir almacenando el valor actual del **EjeY**, por lo que vas a seleccionar el bloque de **Yaccel** directamente haciendo clic en el componente de aceleración.

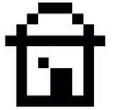
```
cuando AccelerometerSensor1 .CambioEnAceleración
  xAccel yAccel zAccel
ejecutar poner Etiqueta1 . Texto como AccelerometerSensor1 . YAccel
```

Esta secuencia de bloques permitirá mostrar por el Visor el estado actual del sensor. Ahora solo falta actuar dependiendo de qué valor se muestra en cada momento.

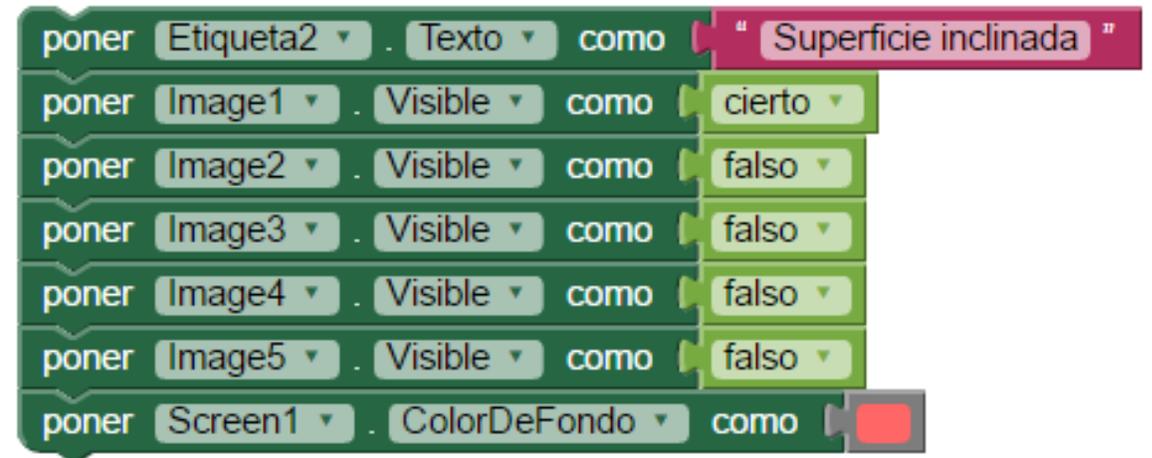


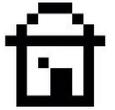
- **Paso 13:** Una vez ya tienes controlado el valor actual del sensor, vas a aplicar varias condiciones de manera que dependiendo del valor se muestra una imagen u otra por el Visor. La condición comprueba si el valor de **Yaccel** es menor o igual a -1, y en caso de ser cierta, ejecuta una serie de bloques que mostrarán una u otra imagen de la animación.





- **Paso 14:** En caso de cumplirse que **Yaccel** es menor o igual que -1, se va a anotar un mensaje en la segunda etiqueta que has agregado antes. Seguidamente colocará como visible la imagen deseada, y como no visible todas las demás, además cambiará el color de fondo de la pantalla para apreciar el resultado de manera rápida.





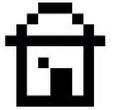
- **Paso 15:** ahora este paso lo deberás repetir cuatro veces más, de manera que **el programa tendrá cinco condiciones**. Dependiendo de que la situación sea una u otra, se van a mostrar las imágenes, el texto y la información que sea adecuada en cada situación.

En las diapositivas siguientes podrás ver cómo queda la estructura de bloques de nuestra App... son básicamente bloques duplicados, ¿se te ocurre cómo ahorrarte trabajo?



```
when AccelerometerSensor1 . AccelerationChanged
  xAccel yAccel zAccel
do
  set Etiqueta1 . Text to AccelerometerSensor1 . YAccel
  if AccelerometerSensor1 . YAccel ≤ -1
  then
    set Etiqueta2 . Text to " Superficie inclinada "
    set Image1 . Visible to true
    set Image2 . Visible to false
    set Image3 . Visible to false
    set Image4 . Visible to false
    set Image5 . Visible to false
  if AccelerometerSensor1 . YAccel ≥ -0.99 and AccelerometerSensor1 . YAccel < -0.5
  then
    set Etiqueta2 . Text to " Superficie inclinada "
    set Image1 . Visible to false
    set Image2 . Visible to true
    set Image3 . Visible to false
    set Image4 . Visible to false
    set Image5 . Visible to false
```

Quinta App: nivel de burbuja



```
if AccelerometerSensor1.YAccel >= -0.49 and AccelerometerSensor1.YAccel < 0.5
then
  set Etiqueta2.Text to "Superficie estable"
  set Image1.Visible to false
  set Image2.Visible to false
  set Image3.Visible to true
  set Image4.Visible to false
  set Image5.Visible to false

if AccelerometerSensor1.YAccel >= 0.51 and AccelerometerSensor1.YAccel < 0.99
then
  set Etiqueta2.Text to "Superficie inclinada"
  set Image1.Visible to false
  set Image2.Visible to false
  set Image3.Visible to false
  set Image4.Visible to true
  set Image5.Visible to false

if AccelerometerSensor1.YAccel >= 1
then
  set Etiqueta2.Text to "Superficie inclinada"
  set Image1.Visible to false
  set Image2.Visible to false
  set Image3.Visible to false
  set Image4.Visible to false
  set Image5.Visible to true
```

