

Capítulo 3

La interfaz de desarrollo



Contenido

1. Introducción
2. Crear un nuevo proyecto. El diseñador
3. La barra de menús de la aplicación
4. La barra de menús del proyecto
5. La paleta de componentes
6. El visor
7. Los componentes
8. Media
9. Las propiedades
10. El editor de bloques
11. Resumen

1. Introducción

Antes de comenzar con los desarrollos, se han de conocer cada uno de los elementos que componen la interfaz de *MIT App Inventor 2*. Conocer al detalle cada uno de ellos otorga cierta seguridad a la hora de emprender la creación de un proyecto.

MIT App Inventor 2 posee dos divisiones claramente diferenciadas, como son el diseñador y el editor de bloques.

A través del diseñador se dará forma a la aplicación, se determinará el aspecto que tendrá posteriormente la interfaz añadiendo y organizando los componentes que en ella incluya.

El diseñador contiene una serie de secciones que se deberán conocer en profundidad para sacar el máximo partido a los componentes de la aplicación.

El editor de bloques permitirá proporcionar la lógica y el comportamiento a los componentes de la aplicación, determinando así la funcionalidad de la misma.

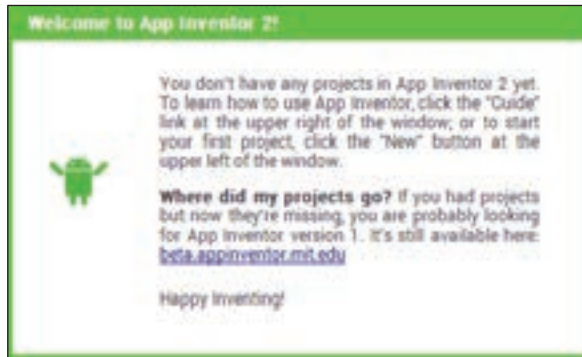
Podrá comprobar que se puede cambiar el idioma del entorno de desarrollo de la aplicación al castellano aunque se ha optado por no hacerlo. El motivo es que toda la documentación e información para la aplicación existente en la red está en inglés y podría crear confusión en las traducciones realizadas en la propia aplicación.

2. Crear un nuevo proyecto. El diseñador

Como se pudo comprobar en el capítulo anterior, la primera vez que se accede a la aplicación la pantalla aparecerá vacía debido a que no se tiene aún ningún proyecto. De hecho, aparece un mensaje de aviso que da la bienvenida a la aplicación. El mensaje advierte que aún no se tiene ningún proyecto, sugiere aprender a usar la aplicación desde la opción *Guide* de la parte superior derecha de la página de inicio, además de invitar a hacer clic sobre el botón

Crea tus aplicaciones Android con App Inventor 2

New Project de la parte superior izquierda de la página y comenzar la creación de un nuevo proyecto.



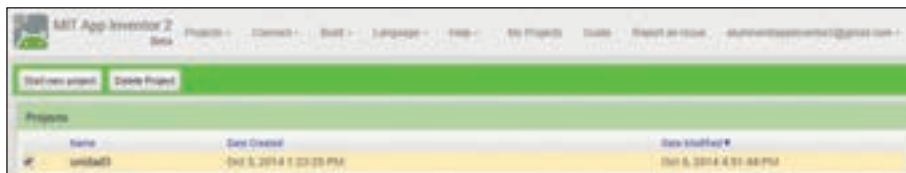
Bienvenida a MIT App Inventor 2

Crear un nuevo proyecto es sencillo. Simplemente se deberá hacer clic sobre el botón New Project, que ya se conoce, y otorgar un nombre al proyecto.



Nombre de nuevo proyecto

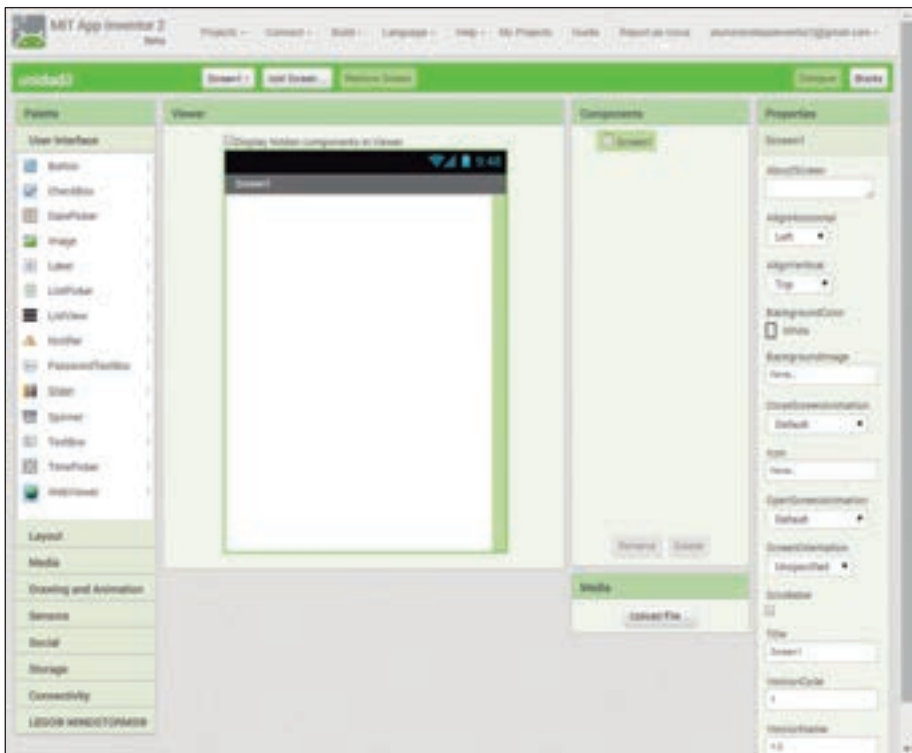
Si se necesita eliminar un proyecto, se puede hacer desde la opción Delete project.



Opciones de proyecto

Después de crear el proyecto, y de manera automática, se abrirá este. Se podrá observar lo que se conoce como el **diseñador** de *MIT App Inventor 2*, con el que se podrá comenzar a desarrollar la aplicación.

Este se compone de una ventana con una serie de barras de menú y paneles que ofrecen todas las herramientas y opciones disponibles en *MIT App Inventor 2* para hacer efectivo el diseño de la aplicación *Android*. A lo largo de este capítulo, se conocerá poco a poco cada uno de los componentes que forman parte del diseñador y se comenzará desde el siguiente apartado con la barra de menús de la aplicación. A través de la siguiente imagen, se pueden observar todos estos componentes en su conjunto.



Diseñador de MIT App Inventor 2

3. La barra de menús de la aplicación

Desde la barra superior de la aplicación se encontrarán diferentes opciones, a modo de menús, con las que se trabajará a lo largo de los desarrollos. Entre ellas se encuentran, a la izquierda de la barra, **Projects**, **Connect**, **Build**, **Language** y **Help**, y a la derecha, **My Projects**, **Guide**, **Report an issue** y la cuenta de usuario de Gmail.

En cada uno de los citados menús se encuentran las siguientes opciones:

3.1. Projects

Engloba las posibles opciones que rodean a la gestión de un proyecto clasificándolas en función de su utilidad. Entre ellas, se tienen las siguientes:

My projects

Permite visualizar su listado de proyectos pudiendo así abrir o consultar el que más interese.

Start new project

Inicia la creación de un nuevo proyecto.

Import Project (.aia) from my computer...

Permite importar un proyecto desde alguna ubicación del ordenador en formato .aia.

Delete project

Elimina un proyecto existente.

Save project

Guarda el estado del trabajo sobre el mismo proyecto en el que se está trabajando.

Save project as

Guarda el estado del trabajo sobre un proyecto diferente al que se está trabajando permitiéndole elegir un nuevo nombre. Después, se continuará trabajando sobre el nuevo proyecto creado.

Checkpoint

Guarda el estado del trabajo permitiendo volver a él en caso de necesitarlo. Se realiza sobre un proyecto diferente al que se está trabajando concediendo la opción de elegir un nuevo nombre para el nuevo proyecto. Esta opción será especialmente útil cuando el desarrollo se encuentre ante alguna decisión difícil y no se esté seguro del resultado. Si la elección no fue la acertada permite recuperar un estado estable.

Export selected Project (.aia) to my computer

Exporta el proyecto seleccionado a una ubicación conocida del ordenador en formato .aia.

Export all projects

Exporta todos los proyectos existentes a una ubicación conocida del ordenador en formato .aia

Import keystore

Permite importar una *keystore*.

Export keystore

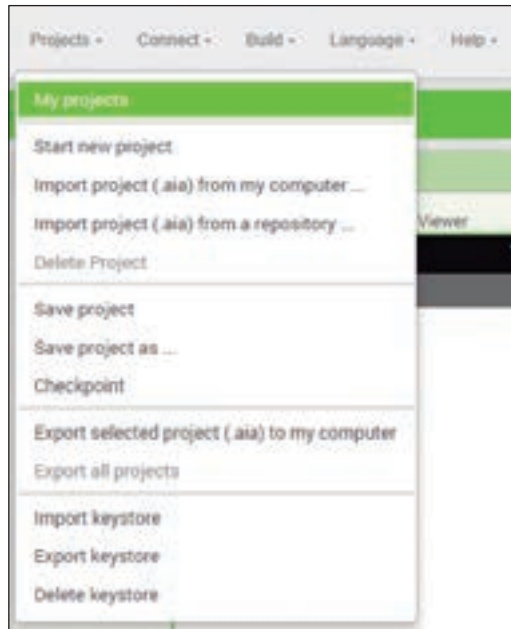
Permite exportar una *keystore*.

Delete keystore

Permite eliminar una *keystore*.

En cuanto a estas últimas opciones, se debe saber que *Android* requiere que las aplicaciones instaladas en un dispositivo estén firmadas por el desarrollador mediante un certificado, también denominado *keystore*, y cuya clave privada corresponde al propio desarrollador, para permitir así su distribución e instalación. Este hecho establece las posibles relaciones de confianza entre las aplicaciones, suministrando las medidas de seguridad necesarias. Google determina una serie de puntos importantes, entre ellos que dicho certificado no sea obligatoriamente firmado por una autoridad certificadora, pudiendo ser creado por el propio desarrollador. Además, un desarrollador debe conocer que la aplicación será firmada para poder ser testeada y depurada con una clave de depuración especial sustituyendo esta al final del desarrollo por una clave privada adecuada para su distribución. Una aplicación deberá estar firmada correctamente para ser instalada en un emulador o dispositivo. Una vez creado el certificado, se podrá emplear cada vez que exporte sus aplicaciones. Otro motivo importante del por qué se firma una aplicación es para que solo el desarrollador de la aplicación la pueda actualizar y/o modificar.

MIT App Inventor 2 genera una *keystore* para que, después de crear una nueva aplicación, la pueda exportar e instalar en un dispositivo *Android*. Si por cualquier motivo se desea utilizar una *keystore* propia, se puede sustituir subiéndola a la aplicación y sobrescribiendo la existente.



Opción de menú Projects



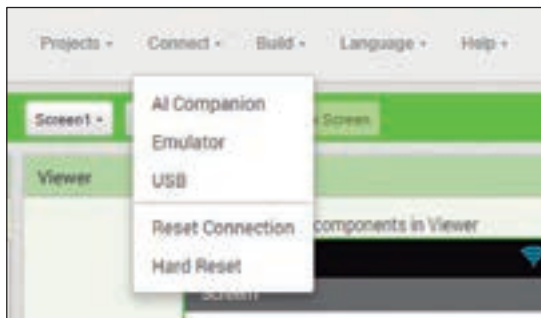
Nota

Si se desea conocer más información sobre las *keystore* y su uso, acceder a la siguiente dirección: <http://developer.android.com/tools/publishing/app-signing.html>

3.2. Connect

El menú **Connect** agrupa las opciones que necesita para probar su aplicación. Entre las opciones que dispone, se ofrece **AI Companion**, una aplicación que se podrá descargar en el dispositivo móvil *Android* desde Google Play de manera gratuita y que, a través de la tecnología QR, permite acceder a la aplicación que se está desarrollando. El nombre completo de la aplicación es *MIT AI2 Companion*. Además, se ofrece un emulador desde el que se podrá, del mismo modo, probar la aplicación. Por último, se podrá hacer uso del dispositivo *Android* real

para probar las aplicaciones conectándolo al ordenador a través del cable USB. Las diferentes opciones de este menú facilitan realizar pruebas y testeos de las aplicaciones en busca de fallos, además de poder comprobar de una manera fiel que tienen el aspecto que deben tener y que realizan las funciones para las que se está desarrollando.

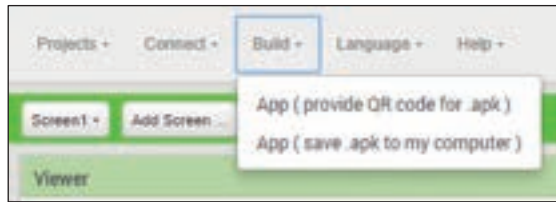


Opción de menú Connect

3.3. Build

Una vez finalizado el desarrollo de la aplicación, se deberá empaquetar para probarla. Este proceso realizará la necesaria compilación y generará el fichero “apk” correspondiente para su instalación en un dispositivo real. Para ello se puede optar por alguna de las dos opciones que se ofrecen en el menú **Build**. La primera de ellas, **App (provide QR code for .apk)**, realiza la acción de compilar la aplicación y muestra un código QR que se puede escanear desde la aplicación *MIT AI2 Companion*. Una vez escaneado el código, la **AI Companion** proporcionará una URL que, insertándola en un navegador, permitirá la descarga del fichero “apk” de la aplicación. El código QR mostrado será válido durante dos horas.

De otro modo, la segunda de ellas, **App (save .apk to my computer)**, realizará la compilación y descarga del fichero “apk” correspondiente a su aplicación directamente al ordenador. El fichero “apk” generado se guardará en la carpeta de descargas.



Opción de menú Build

3.4. Language

Permite la modificación del idioma del interfaz de *MIT App Inventor 2* entre varios posibles.



Opción de menú Language

3.5. Help

Este menú engloba las diferentes opciones de soporte y ayuda necesarias para resolver los diferentes inconvenientes que puedan surgir cuando se trabaje con *MIT App Inventor 2*. Si se recuerda el capítulo anterior, a algunos de estos apartados se puede acceder desde la página de inicio de la aplicación a través de la dirección <http://appinventor.mit.edu/>. Entre las diferentes opciones, se podrán encontrar las siguientes:

About

Muestra los detalles sobre la versión de MIT App Inventor con la que se está trabajando.

Library

Accede al apartado de documentación y soporte de *MIT App Inventor 2* desde los que se podrá consultar cómo resolver las diferentes situaciones que pueden surgir durante el trabajo con la aplicación.

Get Started

Acceso directo al apartado de introducción a *MIT App Inventor 2*. En dicho apartado, se encontrarán instrucciones de instalación, una primera vista tanto del diseñador como del editor de bloques, así como una serie de tutoriales para principiantes y se conocerá cómo empaquetar y compartir las aplicaciones.

Tutorials

Al igual que con las opciones anteriores, **Tutorials** permite ingresar en el apartado del mismo nombre que se ofrece en la página de inicio de la aplicación. En él se obtendrá una serie de instrucciones paso a paso sobre el desarrollo de diferentes aplicaciones de ejemplo. Se podrá encontrar desde algunas básicas hasta otras más complejas.

Troubleshooting

A través de esta opción se brinda la posibilidad de resolver problemas. Se muestran una serie de enlaces rápidos de errores e incidencias comunes que dirigen a sus resoluciones, así como de otros errores que se pueden originar en la aplicación.

Forums

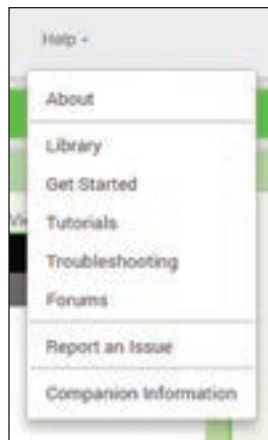
Se trata de una comunidad en la que los usuarios pueden hacer preguntas a los demás con el objetivo de resolver sus propios problemas. Además de pedir ayuda, un usuario podrá ofrecer consejos a otros usuarios. Un foro tiene una serie de normas que se deben respetar a la hora de hacer uso de él. Al ayudar a alguien se debe ser lo más claro y conciso posible, ofreciendo trozos de código, ficheros fuente o capturas de pantalla.

Report an Issue

Al igual que cualquier aplicación en fase de pruebas, *MIT App Inventor 2* puede contener errores. En caso de encontrar alguna incidencia, se puede reportar un informe desde la opción **Report an Issue**. Antes de reportar un error, se ofrecen un conjunto de problemas comunes para el caso en el que este ya haya sido reportado con anterioridad ofreciendo la solución. Si fuese así, se podrá rellenar un formulario con la información del error, del sistema y una dirección de correo electrónico con la que poder contactar con usted.

Companion Information

Permite la descarga de la aplicación *AI Companion* mediante una URL o código QR.



Opción de menú Help

A continuación, se mostrarán las opciones de la barra de menús de la aplicación colocadas en la esquina superior derecha. La mayoría de ellas son accesos directos a opciones que ya se han definido anteriormente pero que se ubican aquí para aumentar su visibilidad y rápido acceso.

3.6. My Projects

Al igual que la opción del mismo nombre del menú **Projects**, permite visualizar el propio listado de proyectos, pudiendo así abrir o consultar el que más interese.



Opción **My Projects** de la barra de menús de MIT App Inventor 2

3.7. Guide

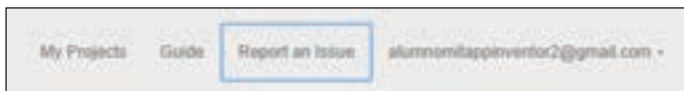
La opción **Guide** dirige al usuario del mismo modo que la opción **Tutorials** del menú de inicio hacia un conjunto de guías de documentación y soporte de *MIT App Inventor 2* desde los que se podrá consultar cómo resolver las diferentes situaciones que pueden surgir durante el trabajo con la aplicación.



Opción **Guide** de la barra de menús de MIT App Inventor 2

3.8. Report an issue

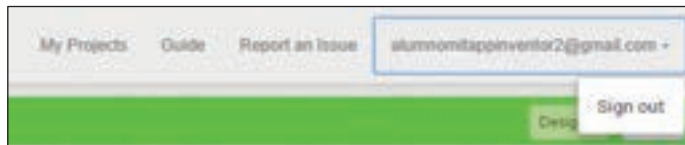
Permite reportar los errores que se encuentren durante el trabajo con *MIT App Inventor 2* y que se piense que no han podido ser detectados.



Opción **Report an Issue** de la barra de menús de MIT App Inventor 2

3.9. Cuenta de usuario

Desde esta opción se podrá cerrar la aplicación que se ha iniciado con la cuenta de usuario de Google, primero haciendo clic sobre el nombre de la cuenta y, acto seguido, sobre la opción **Sign out**.



Opción Cuenta de usuario de MIT App Inventor 2

4. La barra de menús del proyecto

En esta barra se localizan una serie de opciones propias de cada proyecto, como las relativas a las diferentes pantallas que contenga la aplicación, además de ofrecer la posibilidad de cambiar el espacio de trabajo entre el diseñador y el editor de bloques. En ella se ubican una serie de botones, entre los que se encuentran los siguientes:

- **Screen1:** muestra la pantalla actual, si existen varias pantallas aparecerán en este desplegable y se podrá elegir cuál de ellas mostrar en cada momento.
- **Add Screen:** MIT App Inventor 2 permite la creación de múltiples pantallas. Mediante la opción **Add Screen** se podrá generar una nueva pantalla para su aplicación. Por defecto, el proyecto se creará con una pantalla.
- **Remove Screen:** al igual que se permite la creación de nuevas pantallas, se podrá eliminar las que no se utilicen. Esta opción solo estará disponible en caso de existir más de una pantalla en el proyecto. Es indispensable la existencia de una pantalla.
- **Designer:** permite abrir el diseñador en la pantalla de MIT App Inventor 2 para realizar el diseño de las pantallas de la aplicación. Si ya está abierto se encontrará deshabilitado.
- **Blocks:** permite abrir el **Editor de bloques** en la pantalla de MIT App Inventor 2. Tiene por objetivo desarrollar la lógica de la aplicación, la

cual determinará el comportamiento de la misma. Si ya se tiene abierto, se encontrará deshabilitado.



Barra de menús de proyecto

5. La paleta de componentes

Los diferentes elementos que se pueden utilizar en la aplicación se encontrarán en la paleta de componentes. Esta se encuentra localizada en la parte izquierda de la pantalla a modo de columna y dividida en diferentes secciones, agrupando componentes relacionados. La paleta muestra solo una de sus secciones a la vez. Al seleccionar otra, se cerrará la que estuviera abierta para pasar a mostrar los componentes de la que se acaba de elegir. A continuación, se conocerán, mediante una breve descripción, los componentes de cada una de las secciones de la paleta.

5.1. User Interface

Los componentes elementales de uso común que pueden formar parte en cualquier aplicación se encontrarán en esta sección. La relación completa de componentes y su descripción se podrá encontrar en el siguiente listado:

Botón

Permite al usuario interactuar con la aplicación desencadenando alguna acción asociada al botón.

Casilla de validación

Componente *CheckBox*, cuya funcionalidad es la de permitir su selección. Es útil para recoger la elección del usuario entre una lista de opciones o responder con alguna acción a su elección.

Reloj

Si se necesita controlar de alguna forma el tiempo en la aplicación, puede servir de ayuda el componente **Clock**, además de permitir mostrar la hora actual.

Imagen

Cualquier imagen que se quiera incluir a la aplicación se podrá subir a *MIT App Inventor 2* desde las propiedades del componente **Image**.

Etiqueta

Se emplea para presentar en pantalla algún texto. Entre sus usos más comunes se tienen el de poner nombre a los componentes ubicados en la pantalla o para especificar títulos. El nombre de este componente es **Label**.

Lista de elementos

Su uso permite al usuario elegir una opción entre un conjunto de ellas predefinidas de antemano. Cada una de estas opciones incluidas en el componente **ListPicker** será del tipo cadena de texto.

Notificador

En ocasiones, la aplicación puede requerir mostrar avisos o alertas al usuario. De igual modo, este podrá responder a las mismas. El componente no visible **Notifier** permite mostrar alertas y recoger las réplicas del usuario.

Caja de texto para contraseñas

El componente **PasswordTextBox** permite la introducción de una contraseña al usuario ocultando sus caracteres mediante círculos.

Barra de progreso con desplazamiento

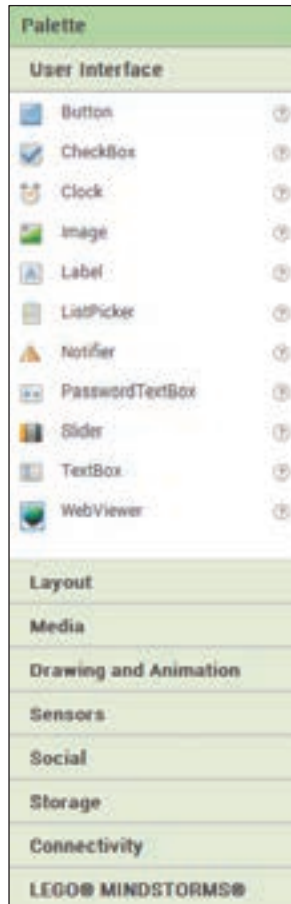
Se refiere al componente **Slider**, el cual permite al usuario modificar el porcentaje de progreso de una barra a través de un pequeño tirador colocado encima de esta.

Caja de texto

TextBox es un componente de entrada de texto al igual que **PasswordTextBox**, pero en este caso se mostrarán los caracteres introducidos en este.

WebView

Este componente hace posible abrir páginas web desde la aplicación. Dispone de una propiedad denominada *HomeUrl*, que permite especificar la URL que se desee abrir. Hay que tener en cuenta que no se dispone de la total funcionalidad de un navegador.



Paleta de componentes User Interface

5.2. Layout

Son componentes que engloban a otros componentes determinando la distribución de estos en la pantalla. Entre ellos, se tienen los siguientes:

Alineación horizontal

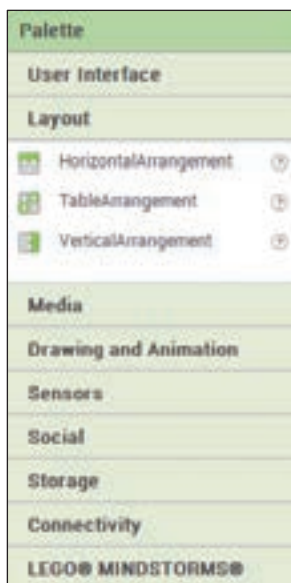
El componente **HorizontalArrangement** dispone los componentes que en él se ubiquen de manera horizontal y de izquierda a derecha. Entre las propiedades que se pueden definir en este componente se encuentran la visibilidad, la altura y la anchura del mismo.

Alineación vertical

Similar al componente anterior, **VerticalArrangement** dispone los componentes que en él se ubiquen de manera vertical y de arriba a abajo. De igual modo, las propiedades que se pueden definir en este componente se encuentran la visibilidad, la altura y la anchura del mismo.

Alineación en tabla

Si se necesita una alineación tabular para los componentes de la aplicación, **TableArrangement** dispone los componentes ubicados en él organizados en filas y columnas. Cada celda será ocupada por un único componente. Entre las propiedades que se pueden definir en este componente se encuentran la visibilidad, el número de filas y columnas que contiene, además de la altura y la anchura de la tabla.



Paleta de componentes Layout

5.3. Media

Estos componentes, que se podrán añadir a la aplicación, dotarán a esta de diferentes comportamientos multimedia. Entre ellos, se encuentran los siguientes:

Grabador de vídeo

Hacer uso de la cámara del dispositivo es algo que se podrá realizar mediante el componente **Camcorder**. Este permitirá que se puedan realizar grabaciones de vídeo desde la aplicación.

Cámara fotográfica

Al igual que ocurre con el vídeo, tomar una fotografía es una funcionalidad que se puede añadir a la aplicación. En este caso, a través del componente **Camera**. Tanto **Camcorder** como **Camera** son componentes no visibles.

Selector de imágenes

El componente **ImagePicker** permite seleccionar una imagen desde la galería de imágenes tras pulsar el botón que representa al componente. Se trata de un componente no visible.

Reproductor

El componente no visible **Player** reproduce sonido o vídeo desde la aplicación, además de controlar la vibración del dispositivo.

Reproductor de sonido

El componente no visible **Sound** reproduce archivos de sonido desde la aplicación, además de controlar la vibración del dispositivo. La diferencia principal entre **Player** y **Sound** es que el primero es más eficiente para sonidos de larga duración y el segundo de ellos, para sonidos cortos.

Grabador de sonido

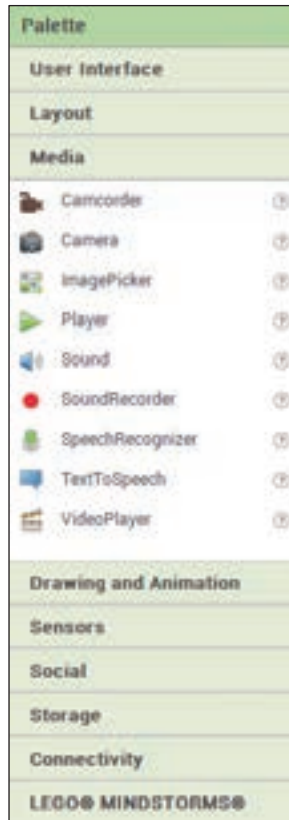
El componente multimedia **SoundRecorder** es un componente no visible que permite la grabación de sonido. No dispone de propiedades pero sí de unos métodos y eventos con los que realizar las grabaciones.

Reconocedor de voz y sintetizador de voz

SpeechRecognizer y **TextToSpeech** son dos componentes de voz orientados, el primero, a convertir el sonido del habla del usuario a texto, y el segundo, a traducir un texto determinado a sonido entendible en el idioma especificado. Para hacer uso del componente **TextToSpeech** se deberá tener configurado en el dispositivo un motor de síntesis de voz.

Reproductor de vídeo

El componente **VideoPlayer** reproduce archivos de vídeo desde la aplicación, añadiendo a ella un área donde se apreciará la imagen. Se permite acceder a los controles de reproducción del componente tocándolo simplemente.



Paleta de componentes Media

5.4. Drawing and Animation

Este conjunto de componentes proporciona a la aplicación dotes de animación, permitiendo interactuar con ellos.

Bola

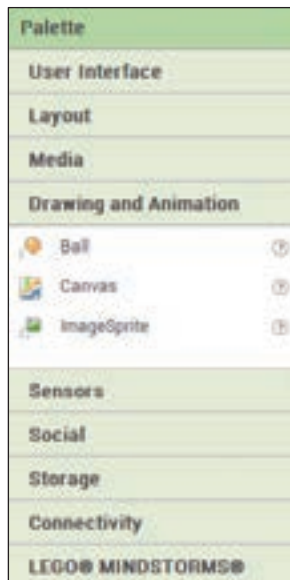
El componente **Ball** es un objeto animado cuya funcionalidad se lleva a cabo sobre un lienzo o componente **Canvas** y reproduce una bola en movimiento.

Lienzo

El componente **Canvas** es un lienzo o panel con el que se podrá dibujar en la pantalla del teléfono o añadir otros componentes.

Imagen animada

ImageSprite es un componente de similar naturaleza a **Ball** con la diferencia de que se puede definir por el usuario la imagen que determine su aspecto. Ambos componentes permiten la interacción entre ellos, factor que facilita una implementación que se base en la lógica de colisiones para la aplicación.



Paleta de componentes Drawing and Animation

5.5. Sensors

Componentes con los que se podrá interactuar con los sensores y demás *hardware* del dispositivo.

Acelerómetro

El componente **AccelerometerSensor** permite, mediante el sensor acelerómetro del terminal, determinar la aceleración producida en el mismo en cada uno de los ejes X, Y y Z, además de detectar cuando es agitado.

Lector de código de barras

BarcodeScanner es un componente no visible que permite interpretar códigos de barras y códigos QR a través de la cámara del dispositivo. Para ello, se deberá tener instalada además la aplicación **Barcode Scanner** de XZing.

Posicionamiento

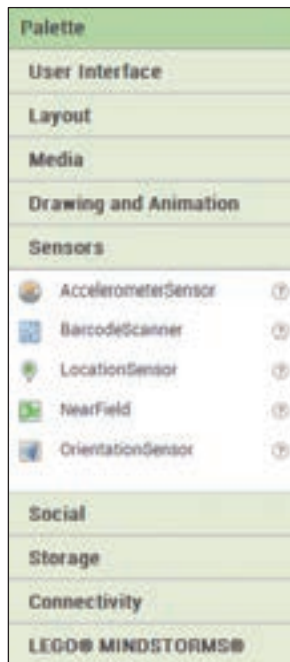
LocationSensor es el componente no visible que se puede añadir a la aplicación para determinar la ubicación del dispositivo. Entre las propiedades que se pueden manipular están la longitud, latitud, altura y dirección. Para hacer uso de este componente el dispositivo deberá poseer la conexión necesaria para localizar la posición y estar activada. Esta puede ser mediante GPS o WI-FI, entre otras.

Comunicación de campo cercano

NearField es un componente no visible que proporciona funcionalidad NFC a la aplicación siempre que sea compatible con el dispositivo con el que se utilice. Es útil para leer y escribir etiquetas, por el momento, de texto. Para ello, es imprescindible tener marcada la propiedad **ReadMode**.

Orientación

El componente no visible **OrientationSensor** determina la orientación del dispositivo. Es posible conocer los valores en ángulos de las propiedades azimut, inclinación y giro del dispositivo.



Paleta de componentes Sensors

5.6. Social

El conjunto de componentes *Social* facilita la integración de la aplicación con diferentes herramientas como contactos, email, llamadas, mensajes o la red social *Twitter*.

Selector de contactos

ContactPicker es un componente que, a través de la pulsación de un botón, permite elegir un contacto de la lista del dispositivo con la finalidad de poder manipular información, como el nombre del contacto o la imagen que tenga establecida.

Selector de email

El componente **EmailPicker** es similar a una caja de entrada de texto, con la salvedad de contener la funcionalidad extra de proporcionar al usuario diferentes

alternativas para autocompletar el texto que está introduciendo. Cabe la posibilidad de que este componente no sea compatible con algunos terminales.

Llamada telefónica

Si la aplicación requiere proporcionar acceso fácil al usuario para que realice llamadas telefónicas, **PhoneCall** permite especificar un número de teléfono y, mediante la pulsación de un botón, realizar una llamada. Se trata de un componente no visible.

Elegir número de teléfono

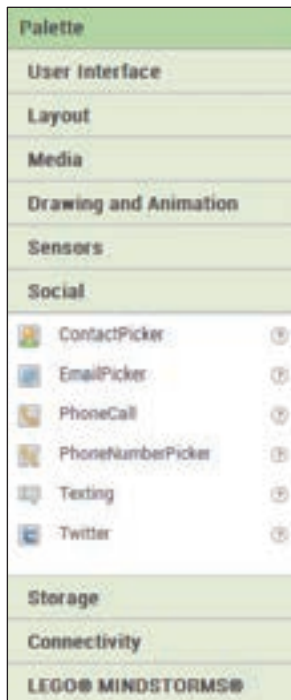
Seleccionar un número de la agenda del teléfono es algo sencillo. El componente **PhoneNumberPicker** permite hacerlo con el fin de recuperar datos como el nombre, número de teléfono o imagen del contacto seleccionado.

Mensajes de texto

El componente **Texting** permite el envío y recepción de mensajes de texto. Posee propiedades que determinan tanto el número de teléfono como el mensaje a enviar. Se trata de un componente no visible.

Twitter

El último de los componentes sociales de los que se dispone en *MIT App Inventor* es el componente *Twitter*. Con él se podrá añadir la funcionalidad propia de la red social de modo que se podrá, entre otras cosas, enviar mensajes a usuarios de la red, seguir a usuarios, obtener listas de seguidores o menciones. Para hacer uso de este componente es necesario que se especifique el valor de sus dos propiedades, **ConsumerKey** y **ConsumerSecret**, con el objetivo de registrar la aplicación en *Twitter* y operar con su usuario. Se trata de un componente no visible.



Paleta de componentes Social

5.7. Storage

Esta sección contiene una serie de componentes para trabajar con datos, facilitando así diferentes métodos de almacenamiento de los mismos.

File

Se hará uso del componente no visible **File** cuando se necesite almacenar información o recuperarla de ficheros que se encuentren en el dispositivo. Se encontrarán métodos en el componente que permitirán determinar el texto que se desee guardar o añadir al fichero, recuperarlo e incluso eliminar dicho fichero. Para guardar información en un fichero se deberá incluir la barra inclinada delante del nombre del mismo. De este modo, el fichero *“/prueba.txt”* se almacenará en la memoria del dispositivo en la ruta *“/sdcard”* por defecto y cuya ruta quedará como *“/sdcard/prueba.txt”*.

Google Fusion Tables

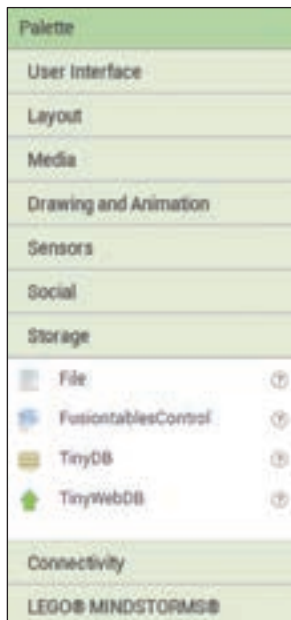
El componente **FusiontablesControl** facilita la comunicación con **Google Fusion Tables**, permitiendo consultar, crear y modificar las tablas que se encuentren en la aplicación de Google. Se trata de un componente no visible que contiene las propiedades **ApiKey**, para vincular el componente a la tabla en cuestión, y **Query**, en la que se especificará la consulta a realizar.

TinyDB

Si se necesita almacenar datos persistentes en la aplicación, es decir, datos que estén disponibles en ella cada vez que la utilice, el componente **TinyDB** ofrece esta funcionalidad. Los datos almacenados a través de este componente no serán efectivos desde el emulador o desplegando la aplicación en el teléfono desde *MIT App Inventor* durante su desarrollo. Para ello se deberá empaquetar la aplicación y descargarla en el terminal. Además, este es un componente no visible y no se muestra en el visor, al ser añadido debajo de este en una sección dedicada a componentes no visibles.

Conexión con servicio web

El componente visible **TinyWebDB** proporciona una conexión con un servicio web, el cual permite almacenar información y consultarla a modo de pares de claves, es decir, se especifica un nombre y su valor. La única propiedad que posee este componente es la dirección del servicio url, cuyo valor es predefinido por *MIT App Inventor 2* a <http://appinvtinywebdb.appspot.com>.



Paleta de componentes Storage

5.8. Connectivity

Este grupo de componentes facilita los mecanismos necesarios para iniciar una serie de acciones a modo de actividades, además de poder realizar comunicaciones a través de Bluetooth o la web.

Iniciador de actividad

El componente no visible **ActivityStarter** permite iniciar desde la aplicación otras que se encuentren instaladas en el teléfono, así como las preinstaladas en el mismo. Estas se determinan por sus actividades que son cada una de las pantallas que posee una aplicación o clases Java. Decir que una aplicación poseerá como mínimo una actividad. En función de la actividad que se pretenda lanzar, se deberán completar las propiedades necesarias del componente **ActivityStarter** correctamente.

Cliente y servidor de Bluetooth

BluetoothClient y **BluetoothServer** son componentes que permitirán a la aplicación comportarse como cliente y servidor Bluetooth.

Comunicación web

Web, componente no visible que proporciona comunicación mediante peticiones HTTP, tanto GET como POST.



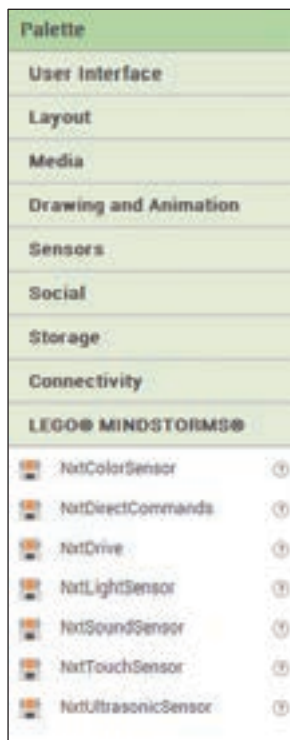
Paleta de componentes Connectivity

5.9. Lego® Mindstorms®

Este conjunto de componentes facilita la integración de aplicaciones *Android* con el juego **Legó Mindstorm** mediante Bluetooth, orientado al aprendizaje de ingeniería robótica, cuyo ámbito de uso se expande desde edades tempranas hasta entornos profesionales. De aquí hay que quedarse solo en el concepto general de estos componentes, sin pormenorizar sus características,

dado que sería necesario poseer dicho juego y sus accesorios para analizar en detalle cada uno de los componentes de la paleta *LEGO® MINDSTORMS®*. Los componentes que *MIT App Inventor* proporciona para establecer la conexión entre **Legó Mindstorm** y una aplicación *Android* son los siguientes:

- **NxtColorSensor.**
- **NxtDirectCommands.**
- **NxtDrive.**
- **NxtLightSensor.**
- **NxtSoundSensor.**
- **NxtTouchSensor.**
- **NxtUltrasonicSensor.**



Paleta de componentes LEGO® MINDSTORMS®



Actividades

1. Navegue por la paleta de componentes con el objetivo de familiarizarse con las categorías y los componentes que se encuentran en ellas.
-

6. El visor

Hecha una primera toma de contacto con los componentes que se pueden usar en las aplicaciones *Android*, el siguiente paso es conocer el panel donde se colocarán para dar forma a la interfaz de usuario de las mismas. Este panel se denomina *Viewer* o **Visor** y el modo para añadir los componentes es sencillo: solo se deben arrastrar desde la paleta correspondiente al visor y colocarlos en la posición deseada. Obsérvese en la siguiente imagen un ejemplo del visor de una aplicación que recoge información y la almacena a través del componente **TinyWebDB**. En ella se pueden diferenciar algunos componentes visibles como **Label**, o etiquetas de texto, **TextBox**, o cajas de texto, y **Button**, o botones, además del componente no visible **TinyWebDB**. Al crear un nuevo proyecto, el visor solo contendrá un componente especial, que agrupará a los demás que posteriormente añada. Este es el componente **Screen** o pantalla. En las aplicaciones se podrán crear nuevas pantallas desde la barra de menús y agrupar sus componentes.



Recuerde

Como mínimo, un proyecto deberá tener una pantalla o **Screen** pero podrá crear las que necesite para su aplicación.



Ejemplo del Visor de MIT App Inventor

Durante el movimiento de arrastre de un componente visible aparecerá una franja de color azul, al entrar en la pantalla de la aplicación, que especificará la ubicación del componente que se pretende añadir con respecto a los componentes que ya existan. En la siguiente imagen, se observa un ejemplo de colocación de un componente en la pantalla, en concreto, se pretende ubicar un componente **TextBox** entre dos componentes **Label**.



Añadir componentes al Visor

Del mismo modo, una vez ubicados los componentes en la pantalla, se podrán mover de sitio arrastrándolos y de nuevo aparecerá la franja azul que especificará la nueva posición.

Por otro lado, en el panel **Visor** se podrá decidir si mostrar o no un componente oculto desde la opción **Display hidden components in Viewer**. Esta opción se verá más clara en adelante cuando se conozca el panel **Propiedades**. Por el momento, saber que desde él se podrán definir componentes como ocultos y, desde la citada opción, mostrarlos en la interfaz para tener una previsualización de su posición en pantalla.



Actividades

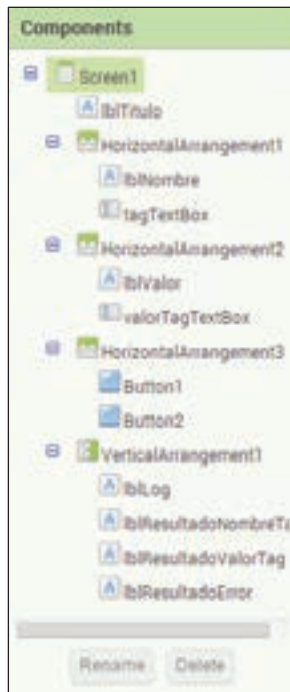
2. Para practicar con la colocación de componentes en el visor, intente reproducir la interfaz que se le ha mostrado con anterioridad a modo de ejemplo para el uso del componente TinyWebDB.
-

7. Los componentes

El siguiente de los elementos a conocer de la ventana del diseñador es el panel de componentes. En él se ubican los componentes que se han ido añadiendo en el visor durante el diseño de la interfaz de usuario de la aplicación, como se ha podido comprobar en el apartado anterior. La organización en la que se presentan los componentes es jerárquica a modo de árbol, encontrándose

primero los componentes situados en función de los valores establecidos para las propiedades **AlignHorizontal** pudiendo ser **Left**, **Center** o **Right** y **AlignVertical** con valor **Top** prestablecido. Al añadir componentes no visibles al visor, estos se mostrarán en la parte inferior del listado del panel **Componentes**.

Desde este panel, se pueden seleccionar los componentes y renombrarlos o eliminarlos del visor. Con esta última opción, desaparecerán de la interfaz gráfica de la aplicación. De la misma forma que ocurre con el panel **Visor**, desde el panel **Componentes**, cuando se selecciona alguno de ellos, se muestran sus características en el panel **Propiedades**, que se verá en detalle en posteriores apartados. Estas características determinarán el comportamiento de dicho componente en la aplicación.



Componentes de un proyecto en el Diseñador

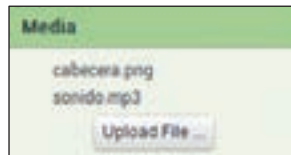


Actividades

3. ¿Cuál es el criterio de ordenación para los componentes que incluye la interfaz gráfica de su aplicación desde el panel Componentes?
-

8. Media

Una aplicación puede hacer uso de diferentes elementos multimedia como imágenes, sonidos o vídeos, en sus formatos compatibles. Estos elementos se podrán subir a los servidores de *MIT App Inventor 2* desde el panel *Media*. Una vez subidos, se mostrarán en él, pudiendo comprobar en todo momento los disponibles. Al seleccionar un componente multimedia, aparecerán dos opciones: la primera de ellas será **Delete**, desde la que se podrá eliminar el componente seleccionado; la segunda es **Download to my computer**, y a través de esta opción se podrá descargar el componente seleccionado al ordenador.



Componentes multimedia de un proyecto



Nota

Los diferentes elementos multimedia que se suban a los servidores de MIT App Inventor no pueden sobrepasar los 5 MB de tamaño.



Actividades

4. Realice la subida de diferentes elementos multimedia a un proyecto y compruebe la cantidad de formatos compatibles que permite MIT App Inventor.
-

9. Las propiedades

En apartados anteriores se ha hablado del panel **Propiedades**. Se trata de un panel en cierto modo dinámico, puesto que, en función del componente seleccionado en el visor o en el panel **Componentes**, mostrará las propiedades específicas de este. Es de esperar que cada componente tenga rasgos particulares y difiera en su comportamiento de otros, aunque en algún caso puedan tener propiedades similares y comunes. Las propiedades de un componente podrán determinar la interactividad de este en la aplicación.

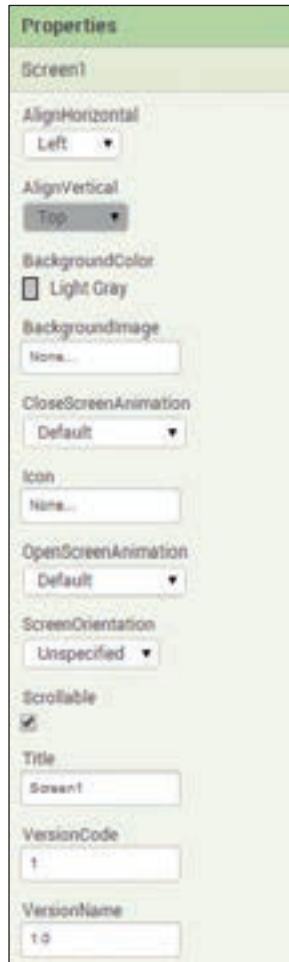
En función de la complejidad de los componentes, estos poseerán una serie de propiedades comunes a otros componentes, cuando son más sencillos, y un conjunto de propiedades específicas que solo poseerán estos, cuando sean más complejos. Por ejemplo, tanto el componente **Image** como el componente **Canvas** poseerán propiedades que determinen sus dimensiones. Estas son **Width** y **Height**. Sin embargo, el segundo de ellos posee la propiedad **BackgroundColor** para determinar el color de fondo del mismo, que en el caso del primero no existe.



Recuerde

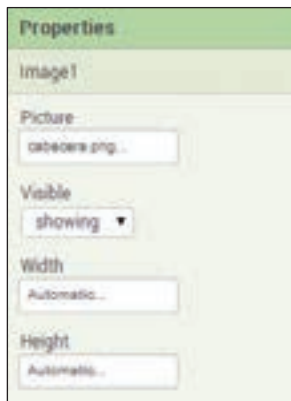
Las propiedades de un componente podrán determinar la interactividad de este en la aplicación.

En la siguiente imagen se podrán observar las propiedades de las que se compone el componente especial **Screen**.



*Propiedades del componente **Screen***

De otro modo, para ver las notables diferencias explicadas con anterioridad en cuanto al número de propiedades de cada componente, basta observar en la siguiente imagen las propiedades pertenecientes al componente **Image**.



Propiedades del componente *Image*

Además de poder modificar las características propias de cada componente desde el panel **Propiedades**, se podrán modificar desde el **Editor de Bloques** en función de la ejecución de ciertas acciones o eventos que se produzcan en la aplicación, como por ejemplo pueden ser la pulsación de un botón o el deslizar el dedo sobre una imagen. El **Editor de Bloques** se verá en el siguiente apartado.



Actividades

5. Añada diferentes componentes al visor y selecciónelos desde la paleta de componentes. Observe cómo cambian sus propiedades.
-

10. El editor de bloques

Durante el inicio de esta unidad se adelantó que *MIT App Inventor 2* se estructuraba en dos grandes grupos: el primero se trata del *diseñador*, del cual se acaban de ver todas sus secciones y paneles; el segundo de ellos es el *editor de bloques*, al que se dedicará este apartado.

El propósito u objetivo del editor de bloques es ensamblar los componentes de la aplicación definiendo la lógica y el comportamiento de los mismos. Se utilizan un conjunto de bloques que corresponden tanto a los componentes que se han añadido durante el diseño de la interfaz desde el panel **Componentes** al visor, así como los que ofrece la propia aplicación. Se podrán especificar eventos, métodos y atributos sobre los componentes de la aplicación a través de los bloques.



Definición

Evento

Es un mecanismo usado en programación para notificar que ha ocurrido algo. Por ejemplo, se ha pulsado una tecla, se ha hecho clic con el ratón o se ha modificado la propiedad de algún componente.

Método

Conjunto de instrucciones agrupadas con una finalidad común y asociadas a un nombre a través del cual se podrán ejecutar. Un método suele ir vinculado a un componente sobre el que realizará alguna acción. Por ejemplo, para una imagen, se define el método `CambiarTamaño` (ancho, alto) mediante el que se podrá definir un nuevo tamaño para la imagen a la que se aplique. Al definir el método habrá que determinar además las instrucciones que harán posible el objetivo del mismo.

Atributo

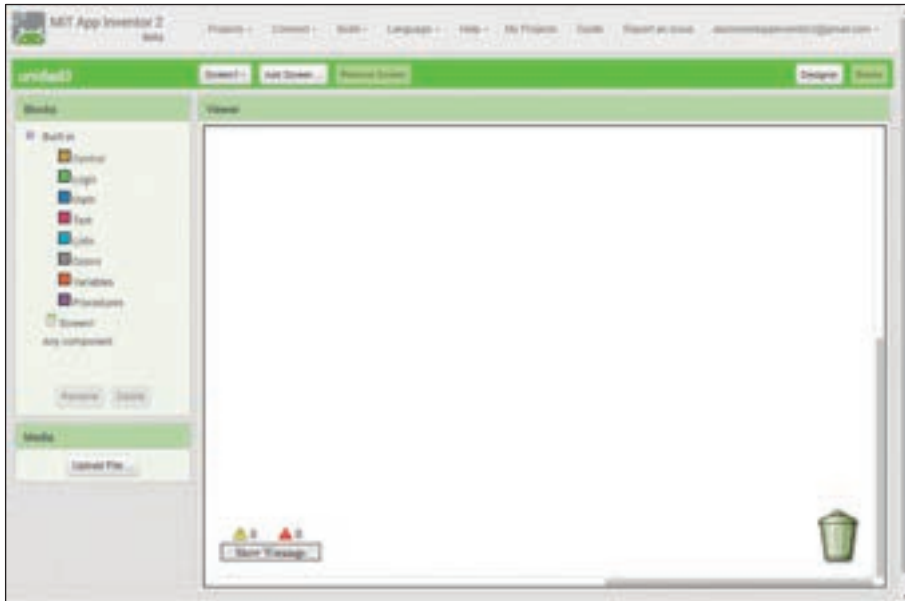
Cada una de las características que diferencian a un componente, determinando su comportamiento, estado, apariencia y demás cualidades que posea. En *Mit App Inventor 2* son las propiedades de cada componente. Un ejemplo puede ser el tamaño o tipo de letra de una etiqueta.

Al editor de bloques se accede desde la barra de menús del proyecto que se tenga abierto pulsando sobre el botón **Blocks**. Al hacer clic sobre este botón, cambiará la apariencia de la aplicación *MIT App Inventor 2*, desapareciendo las secciones propias del diseñador como **Palette**, **Components** y **Properties**. Así como desaparecen unas secciones aparecen otras propias del editor de bloques

Crea tus aplicaciones Android con App Inventor 2

como **Blocks**. Decir que se mantienen tanto la sección del visor o **Viewer**, así como **Media**, siendo comunes a ambos espacios de trabajo.

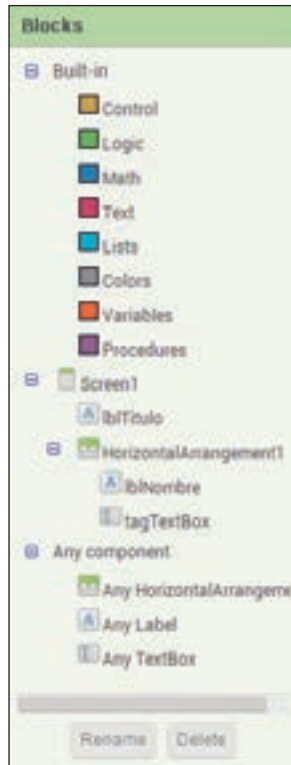
Una vez seleccionado el botón del editor de bloques, la aplicación tendrá el aspecto que se muestra en la siguiente imagen.



El editor de bloques

Como se puede observar, la pantalla de la aplicación se encuentra ocupada en su mayor parte por el visor al lado derecho de esta, dejando el lado izquierdo para los bloques y los recursos multimedia.

En la imagen siguiente se puede apreciar que existen tres apartados que determinan la división de los tipos de bloques como son **Built-In**, **Screen1** y **Any component**.



Panel **Blocks** del Editor de bloques

Aclarar que el inglés utilizado en los bloques que se van a conocer a continuación es sencillo e intuitivo y con la práctica se podrá llegar a aprender su significado sin ninguna dificultad. Si aun así, se observan problemas para la comprensión del significado de alguno de los bloques, se recomienda indagar su significado en un diccionario antes de continuar o en alguna de las herramientas de traducción de internet.

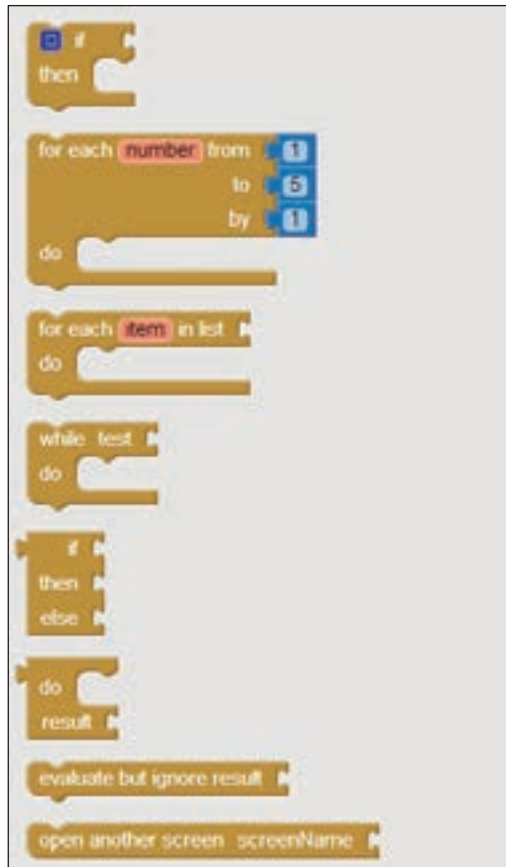


Recuerde

Si se observan problemas por el idioma para la comprensión de alguno de los bloques se recomienda consultar su significado en un diccionario antes de continuar.

En el primer apartado, **Built-In**, se encontrarán los bloques que *MIT App Inventor 2* incorpora para poder construir la lógica de la aplicación. En esta pestaña se encontrarán diferentes categorías que se caracterizarán por el tipo de operaciones a realizar, y que a continuación se muestran:

- **Control:** la categoría **Control** engloba una serie de estructuras de control mediante las que se podrá definir la lógica de la aplicación. Por lógica se entiende el comportamiento que tendrá la misma ante diferentes situaciones. Entre los bloques más representativos están **if**, **ifelse**, **foreach** o **while**. Además hay una serie de bloques de llamada que realizarán una serie acciones predefinidas en cada una de ellas y relacionadas con la pantalla activa y la propia aplicación.



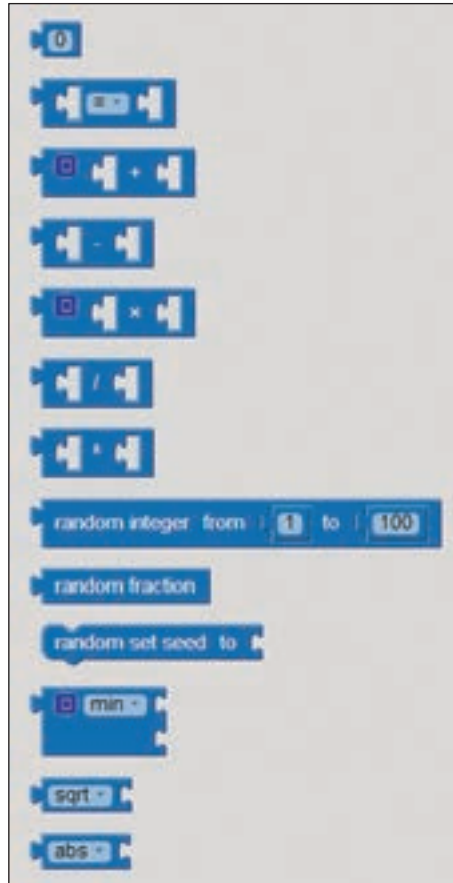
Bloques de la categoría **Control**

- **Logic:** se ofrecen bloques que devuelven los valores lógicos o booleanos *true* o *false*, es decir, *verdadero* o *falso*. Se puede comparar la igualdad de dos valores y evaluar mediante operaciones *and* (y) y *or* (o).



Bloques de la categoría Logic

- **Math:** esta categoría permitirá realizar operaciones matemáticas, entre ellas, la de asignar un valor numérico a una variable, realizar comparaciones entre dos valores numéricos, operaciones matemáticas simples como sumar, restar, multiplicar, dividir o la raíz cuadrada, además de otras más complejas como generar números aleatorios, calcular el cociente o resto entre dos números, así como obtener el valor absoluto de un número, su redondeo o el cálculo de potencias. Por último, también se ofrecen operaciones para realizar cálculos matemáticos de ángulos en grados.



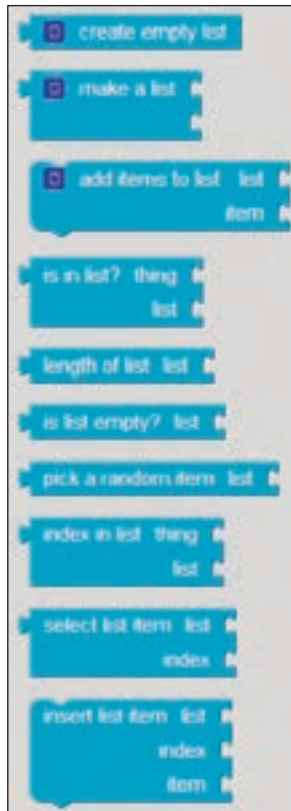
*Bloques de la categoría **Math***

- **Text:** en este caso, se podrán realizar todas las operaciones relacionadas con cadenas de texto. Entre otras, asignar un valor de texto a una variable, unir dos cadenas, calcular la longitud de una cadena, comprobar si se trata de una cadena vacía, eliminar espacios en blanco de una cadena si los hubiera, convertir a mayúsculas/minúsculas, comprobar si alguna cadena contiene algún fragmento de texto específico, dividirla por algún patrón e incluso reemplazar alguna secuencia de caracteres por otra.



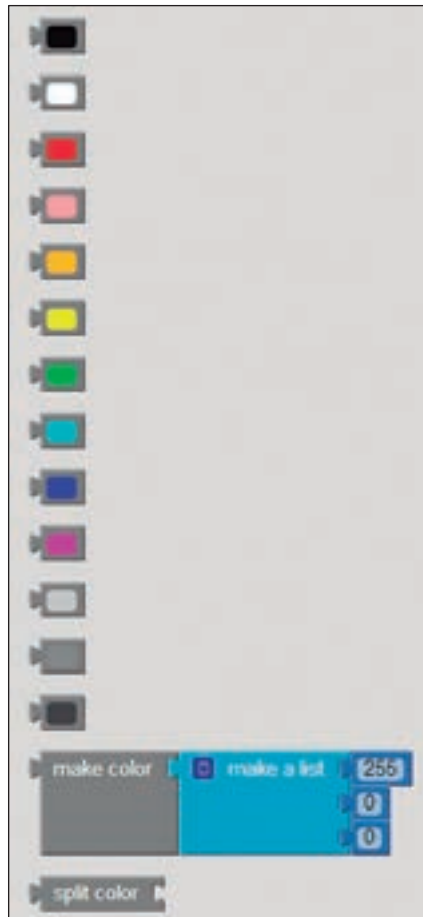
*Bloques de la categoría **Text***

- **Lists:** para el caso de las listas, entre otras operaciones se podrán crear listas, seleccionar elementos incluidos en ellas, reemplazarlos, eliminarlos, insertar nuevos elementos en una posición específica o al final de la lista, conocer la longitud de una lista, buscar elementos en una lista, conocer la posición de un elemento, evaluar si la lista se encuentra vacía o no, realizar una copia completa de una lista o determinar si una variable definida es una lista.



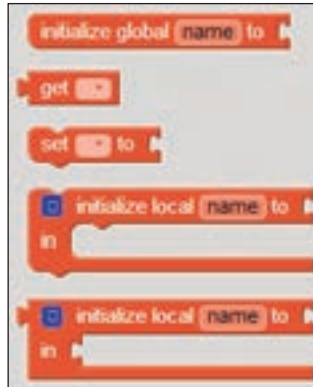
Bloques de la categoría Lists

- **Colors:** desde aquí se podrá hacer uso en las aplicaciones de un conjunto de colores predefinidos, además de poder crear uno mismo a medida los colores que se necesiten proporcionando el código RGB o hexadecimal.



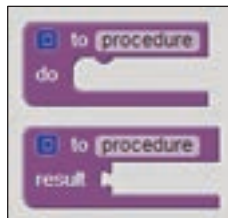
*Bloques de la categoría **Colors***

- **Variables:** esta categoría proporciona los bloques necesarios para definir las variables que se necesiten, inicializarlas o manipularlas.



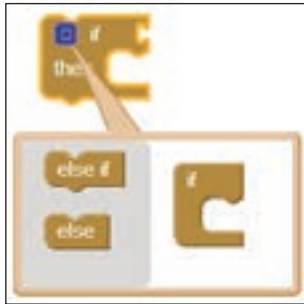
Bloques de la categoría **Variables**

- **Procedures:** del mismo modo, se facilitan los bloques necesarios para definir procedimientos, que podrán devolver algún resultado o no.



Bloques de la categoría **Procedures**

Ya vistos diferentes ejemplos de cada una de las categorías de bloques predefinidos que *MIT App Inventor 2* proporciona, se habrá notado que algunos de ellos contienen un cuadrado azul en su esquina superior izquierda. Si se hace clic encima de él se podrán descubrir diferentes opciones que se pueden agregar al propio bloque. Obsérvese el ejemplo de la siguiente imagen. En él se podrá añadir al bloque lógico **if-then**, las opciones **else if**, **else** e incluso anidar otro bloque **if**. Este recurso es útil a la hora de afrontar un problema dado que ofrece diferentes alternativas para resolverlo.



Más opciones para los bloques

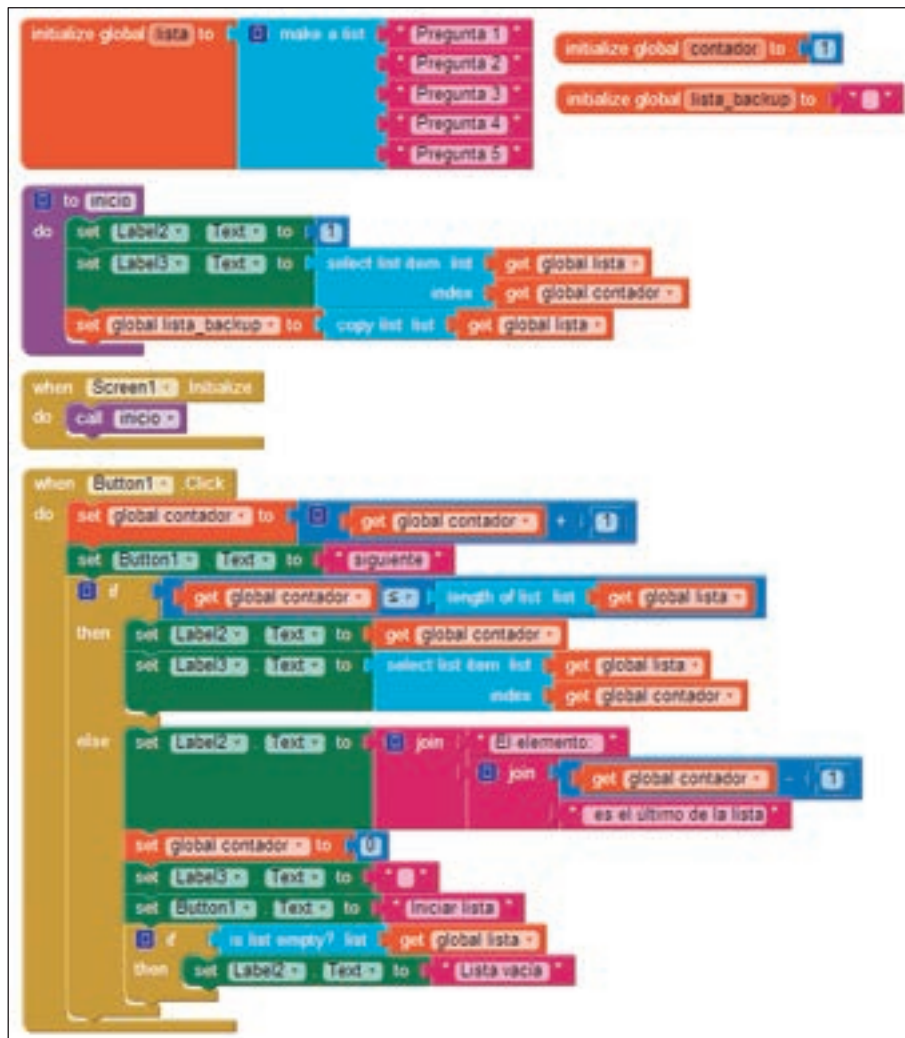
En el segundo de los apartados, **Screen1**, están los componentes que se añadieron a la interfaz de la aplicación. Si la aplicación tiene más de una pantalla, aquí aparecerán los componentes de la pantalla que se tenga en este momento seleccionada en el diseñador. Cada componente posee diferentes bloques que podrán determinar su comportamiento y establecer sus propiedades. Por ejemplo, un componente **Button** dispondrá de bloques encargados de detectar eventos tales como si el botón ha sido pulsado y qué hacer en su respuesta.

Por último, el apartado **Any component** se compone de bloques genéricos, es decir, no se aplican a un botón o etiqueta concreta sino, por ejemplo, a todos los botones o todas las etiquetas respectivamente.

Los diferentes bloques se añadirán seleccionándolos y estos se situarán automáticamente en el panel vacío situado a la derecha, que se pudo observar anteriormente, y no arrastrándolos, al igual que los componentes de la interfaz gráfica al panel visor del diseñador.

En las siguientes imágenes se podrá observar a modo de ejemplo el aspecto de la interfaz gráfica de una aplicación en el panel **Visor**, así como el conjunto de bloques que conforman la lógica de la aplicación. No hay que preocuparse si no se entiende el significado de alguno de ellos. Con la práctica, se irán conociendo poco a poco.





Ejemplo de interfaz gráfica y bloques



Actividades

6. En una actividad anterior se le pidió que añadiera distintos componentes al visor. Acceda al editor de bloques y navegue por los bloques que se proponen para ellos para conocerlos.

11. Resumen

La primera vez que se accede a la aplicación, la pantalla aparecerá vacía debido a que no tiene aún ningún proyecto. Crear un nuevo proyecto es sencillo. Simplemente se deberá hacer clic sobre el botón **New Project** y otorgar un nombre al proyecto. Además, en la barra superior de la aplicación se ofrecen otras opciones agrupadas en los menús **Project, Connect, Build, Help, Language**, a la izquierda, y **My Projects, Guide, Report an issue** y su cuenta de usuario de Gmail, a la derecha. Además, se podrá eliminar un proyecto seleccionado, así como crear otro.

Los diferentes elementos que se pueden utilizar en la aplicación se encontrarán en la **Paleta de componentes** que se encuentra localizada en la parte izquierda de la pantalla a modo de columna y dividida en diferentes secciones, agrupando componentes relacionados.

El panel donde se colocarán los componentes para dar forma a la interfaz de usuario se denomina **Viewer** o **Visor**, y el modo para añadirlos es sencillo, solo se deberán arrastrar desde la paleta correspondiente al visor y colocarlos en la posición deseada.

El cuanto al panel **Componentes**, saber que en él se ubican los componentes que se han ido añadiendo en el visor durante el diseño de la interfaz de usuario de la aplicación. Su organización es jerárquica a modo de árbol. Al añadir componentes no visibles al visor, estos se mostrarán en la parte inferior del listado del panel **Componentes**. Desde este panel se pueden seleccionar los componentes y renombrarlos o eliminarlos del visor. Con esta última opción desaparecerán de la interfaz gráfica de la aplicación.

Se pueden añadir diferentes elementos multimedia, como imágenes, sonidos o vídeos, en sus formatos compatibles, subiéndolos a los servidores de *MIT App Inventor 2* desde el panel **Media**. Posteriormente, se podrá eliminar el componente seleccionado o descargarlo al ordenador.

El panel **Propiedades** es un panel dinámico que mostrará las propiedades específicas del componente seleccionado en el visor o en el panel **Componentes**.

Cada componente tiene rasgos particulares que hace que difiera en su comportamiento de otros, aunque en algún caso puedan tener propiedades comunes.

El objetivo del editor de bloques es ensamblar los componentes de la aplicación y definir la lógica y el comportamiento de la misma. Para acceder al editor de bloques se deberá hacer clic sobre la opción **Blocks** de la barra de opciones del proyecto.