Análisis para el desarrollo móvil con **App Inventor.**

Material
Actividad de aprendizaje 2

Indice

Introducción:

1. Componentes Básicos (Basic).
1.1. Button (Botón).
1.2. Canvas (Lienzo)
1.3. CheckBox (casilla de verificación)
1.4. Clock(Reloj)
1.5. Image (Imagen)
1.6. Label (Etiqueta)
1.7. ListPicker (Lista de selección)
1.8. Screen (Pantalla)
1.9. PasswordTextBox (cuadro de texto para contraseña)
1.10. TextBox
1.11 TinyDB

2. Componentes Multimedia (Media)
2.1. Camera (Cámara)
2.2. ImagePicker ((Selector de imagen)
2.3. Player (Reproductor de audio)
2.4. Sound (sonido)
2.5. VideoPlayer (Reproductor de vídeo)

3. Componentes Animación (Animation)
3.1. Ball (bola)
3.2. ImageSprite (Sprite de imagen)

4. Componentes Sociales (Social)
4.1. ContactPicker (Selector de contactos)
4.2. EmailPicker (Selector de correo electrónico)
4.3. PhoneCall (llamada telefónica)
4.4. PhoneNumberPicker (Selector de número de teléfono)
4.5. Texting (mensajes de texto)
4.6. Twitter

5. Componentes Sensores (Sensor)
5.1. AccelerometerSensor (Sensor acelerómetro)
5.2. LocationSensor (Sensor de posicionamiento)
5.3. OrientationSensor (Sensor de orientación)

6. Componentes Alineamientos de Pantalla
6.1. HorizontalArrangement (Alineamiento horizontal)
6.2. TableArrangement (Alineamiento en tabla)
6.3. VerticalArrangement (Alineamiento vertical)

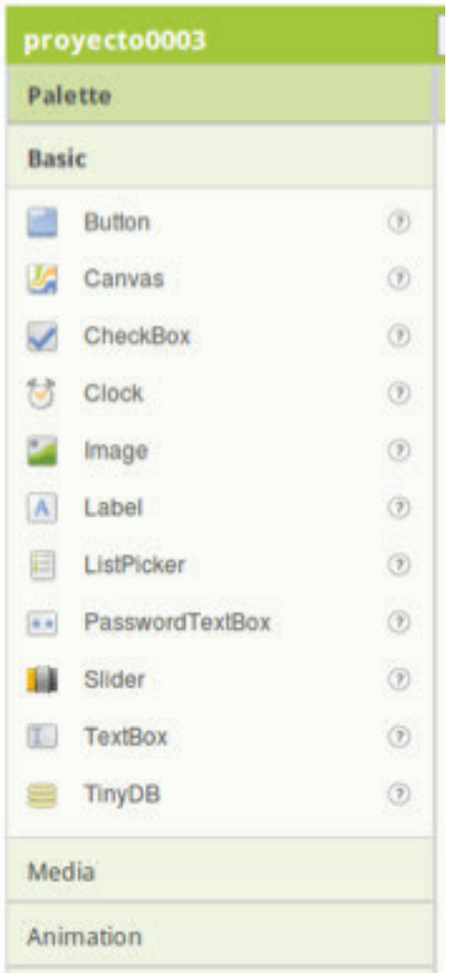
7. Componentes LEGO MINDSTORMS
7.1. NxtDirectCommands
7.2. NxtColorSensor
7.3. NxtLightSensor
7.4. NxtSoundSensor
7.5. NxtTouchSensor
7.6. NxtUltrasonicSensor
7.7. NxtDrive

8. Otros Componentes
8.1. ActivityStarter (Arranque de actividades)
8.2. BarcodeScanner (Lector de código de barras)
8.3. BluetoothClient (Cliente de Bluetooth)
8.4. BluetoothServer (Servidor de Bluetooth)
8.5. Notifier (Notificador)
8.6. SpeechRecognizer (Reconocedor de voz)
8.7. TextToSpeech (Sintetizador de voz)
8.8. TinyWebDB (base de datos web)
8.9. Web

9. Webgrafía

Introducción:

Al crear un nuevo proyecto Android, Google App Inventor nos abrirá la ventana de diseño de la aplicación:



En la parte izquierda tendremos los componentes visuales y no visuales que podremos añadir a nuestra aplicación con tan solo arrastrar y soltar:

A continuación mostramos la lista de componentes disponibles en la paleta (Palette) agrupados por tipo. Hay que mencionar que dicha lista puede variar pues Google App Inventor es una versión Beta y será susceptible de modificaciones futuras añadiendo nuevos componentes y mejorando los actuales:

1.Componentes Básicos (Basic).

1.4. Clock(Reloj)

Se puede utilizar un componente del reloj para crear un contador de tiempo que señala los acontecimientos a intervalos regulares.
El componente de reloj también hace varias conversiones y manipulaciones con unidades de tiempo.
El formato interno usado por el reloj se llama instante. Un instante es un paquete de datos de tiempo diversos, que contiene, entre otros datos, los milisegundos que han pasado desde el inicio de 1970 en el calendario Gregoriano, la zona horaria, el día del año, el día de la semana, el día del mes, el periodo AM o PM, la hora del día, el minuto de la hora, el segundo del minuto, o el milisegundo del segundo.
Uno de los usos de la componente de reloj es el de temporizador: permite configurar un intervalo de tiempo al final de cada uno de los cuales se disparará el evento de temporizador.
Un segundo uso del componente de reloj es el de manipular el tiempo, y expresar el tiempo en varias unidades. . El método Now devuelve el momento actual como un instante. El reloj proporciona diversos métodos para manipular instantes, por ejemplo, devolver el instante correspondiente a varios segundos, meses o años desde un instante dado. También proporciona métodos para mostrar el día, hora, minuto y segundo, ..., de un instante dado.

1.5. Image (Imagen)

Se usa el componente de imagen para representar imágenes que los usuarios seleccionan y manipulan.
Un componente de la imagen muestra una imagen. Se puede especificar la imagen para mostrar y otros aspectos de la apariencia de la imagen en el diseñador o en el Editor de bloques.

1.6. Label (Etiqueta)

Las etiquetas son componentes que se utilizan para mostrar texto.
Una etiqueta muestra el texto especificado en la propiedad Text. Otras propiedades, que se pueden establecer tanto en el diseñador como en el editor de bloques, controlan la apariencia y la colocación de ese texto.

1.7. ListPicker (Lista de selección)

Los usuarios pueden tocar un componente de lista de selección para elegir un elemento de una lista de cadenas de texto.
Cuando un usuario toca el botón selector, se muestra una lista de elementos de texto para que el usuario pueda elegir. Los elementos de texto se puede especificar a través del diseñador o el Editor de bloques mediante el establecimiento de la propiedad ElementsFromString mediante la concatenación de los valores separados por comas (por ejemplo, selección1, selección2, selección3) o mediante el establecimiento de la propiedad Elements de una List (Lista) en el editor de bloques.
Otras propiedades, incluyendo TextAlignment y BackgroundColor , afectan a la apariencia del botón y si puede ser desplegado (Enabled).

1.8. Screen (Pantalla)

La pantalla no aparece en la paleta igual que los restantes componentes, pero aparece automáticamente en el proyecto. Cada proyecto tiene una pantalla, llamada Screen1. Este nombre no puede ser cambiado.

1.9. PasswordTextBox (cuadro de texto para contraseña)

Los usuarios introducen las contraseñas en un componente de cuadro de texto para contraseña, que oculta el texto que se ha escrito en él.
Un cuadro de texto para contraseña es el mismo que el componente TextBox ordinario, excepto en que no muestra los caracteres tecleados por el usuario.
Se puede obtener o establecer el valor del texto en la caja con la propiedad Text. Si Text está en blanco, puede utilizar la propiedad Hint para proporcionar al usuario una sugerencia de lo que puede escribir. La Hint aparece como texto débil en el cuadro.
El componente de cuadro de texto para contraseña se utiliza generalmente con un componente de botón. El usuario pulsa el botón después de la introducción de texto.

1.10. TextBox

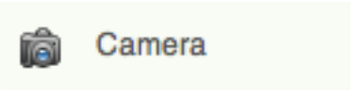
Los usuarios pueden introducir texto en un componente de cuadro de texto.
El valor del texto inicial o el que el usuario haya introducido en un componente de cuadro de texto está en la propiedad Text. Si Text está en blanco, puede utilizar la propiedad Hint para proporcionar al usuario una sugerencia de lo que escribir. La sugerencia aparece como texto débil en el cuadro.
La propiedadl MultiLine determina si el texto puede tener más de una línea. Para una caja de una única línea de texto, el teclado se cerrará automáticamente cuando el usuario presiona la tecla Listo. Para cerrar el teclado para cuadros de texto de varias líneas, la aplicación debe utilizar el método HideKeyboard o confiar en que el usuario pulse la tecla Atrás.
La propiedad NumbersOnly restringe al teclado numérico, de modo que sólo puedan introducirse números.
Otras propiedades afectan a la apariencia del cuadro de texto (TextAlignment , BackgroundColor , etc) y si se puede utilizar (Enabled).
Los cuadros de texto se suelen utilizar con el componente Button , para que el usuario haga clic en el botón cuando se haya completado la entrada del texto.
Si el texto introducido por el usuario no se debe mostrar, usar PasswordTextBox en su lugar.

1.11. TinyDB

Se puede utilizar un componente TinyDB para almacenar los datos que estarán disponibles cada vez que la aplicación se ejecuta. TinyDB es un componente no visible.
Las aplicaciones creadas con App Inventor se inicializan cada vez que se ejecutan. Si una aplicación establece el valor de una variable y el usuario cierra la aplicación, el valor de esa variable no será recordado la próxima vez que se ejecuta la aplicación.
TinyDB es un almacén de datos persistente para la aplicación, es decir, los datos almacenados estarán disponibles cada vez que se ejecute la aplicación. Un ejemplo podría ser un juego que guarda la puntuación más alta, y la recuperaa cada vez que se juega el juego.
Los elementos de datos se almacenan en las etiquetas. Para guardar un elemento de datos, se debe especificar la etiqueta bajo la que debe ser almacenado bajo. Posteriormente, se puede recuperar el elemento de datos que se almacena en una etiqueta determinada. Si no hay ningún valor almacenado en una etiqueta, entonces el valor devuelto es el texto vacío. En consecuencia, para ver si una variable tiene un valor almacenado en ella, comprobar si el valor de retorno es igual a la de texto vacío (es decir, un cuadro de texto sin texto, en blanco).
Sólo hay un almacén de datos por aplicación. Si usted tiene múltiples componentes TinyDB, se utilizará el mismo almacén de datos. Para obtener el efecto de almacenes separados, utilice claves diferentes. Además, cada aplicación tiene su propio almacén de datos. No se puede utilizar TinyDB para pasar los datos entre dos aplicaciones diferentes en el teléfono.

2.Componentes Multimedia (Media)

2.1. Camera (Cámara).

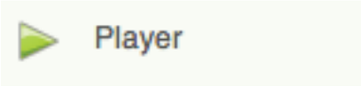


Se puede usar el componente cámara para tomar una foto con el teléfono.
La cámara es un componente no visible, que toma una fotografía con la cámara del dispositivo. Después de tomar la fotografía, la ruta al archivo en el teléfono que contiene la imagen está disponible como un argumento del evento AfterPicture.La ruta puede ser utilizada, por ejemplo, como la propiedad imagen de un componente de imagen.

2.2. ImagePicker (Selector de imagen).

Use un componente selector de imágenes para elegir una imagen de la galería de imágenes.
Un selector de imágenes es una especie de botón. Cuando el usuario pulsa un selector de imágenes, aparece la galería de imágenes del dispositivo, y el usuario puede elegir una imagen. Después de que el usuario elige una imagen, la propiedad ImagePath nos proporciona la ruta de la imagen. A continuación, se puede utilizar ese resultado, por ejemplo, para establecer la imagen de un botón.

2.3. Player (Reproductor de audio).

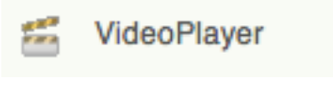


Utilizar un componente player para reproducir un archivo de audio o video, o para que vibre el teléfono.
Player es un componente no visible, que reproduce audio o vídeo y controles de vibración del teléfono. El nombre de un archivo multimedia se especifica en su propiedad origen, que se puede establecer en el Diseñador o en el Editor de bloques. La duración del tiempo de vibración se especifica en el editor de bloques en milisegundos (milésimas de segundo).
Utilizar un componente player para reproducir archivos de sonido largos, archivos de vídeo y hacer vibrar el teléfono. Para reproducir archivos de sonido cortos, tales como efectos de sonido, use un componente sound en su lugar.

2.4. Sound (sonido).

Utilizar un componente de sonido para reproducir un archivo de audio, o para que vibre el teléfono.
El sonido es un componente no visible, que reproduce archivos de sonido y vibra el número de milisegundos (milésimas de segundo) que se indica en el Editor de bloques. El nombre del archivo de sonido se puede especificar en el diseñador o en el Editor de bloques.
Este componente es el mejor para archivos de sonido cortos, tales como efectos de sonido, mientras que el componente player es más eficiente para sonidos largos, tales como las canciones.

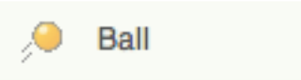
2.5. VideoPlayer (Reproductor de vídeo)



Utilizar un componente VideoPlayer para reproducir un archivo de vídeo.
Reproductor de vídeo es un componente multimedia que reproduce vídeos. Un reproductor de vídeo aparece en su aplicación como un rectángulo. Si el usuario pulsa el rectángulo, los controles multimedia aparecen: reproducción / pausa, saltar hacia delante y saltar hacia atrás. Su aplicación puede controlar el comportamiento de la reproducción mediante una llamada a los métodos Start , Pause , y SeekTo.
Los archivos de vídeo debe estar en Windows Media Video (. Wmv), 3GPP (.3 gp), o MPEG-4 (. Mp4).
App Inventor acepta archivos de vídeo de hasta 1 MB de tamaño y limita el tamaño total de una aplicación a 5 MB, por lo que no todos los archivos de video pueden utilizarse. Si sus archivos de medios de comunicación son demasiado grandes, es posible que obtenga errores al empaquetar o instalar la aplicación, en cuyo caso se deben reducir el número de archivos multimedia o sus tamaños. Ciertos software de edición de vídeo, como Windows Movie Maker o iMovie de Apple, puede ayudar a reducir el tamaño de los vídeos, acortándolos o volviendo a codificarlos en formatos más compactos.

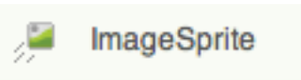
3.Componentes Animación (Animation)

3.1. Ball (bola)



Un componente de bola es un tipo particular de sprite (objeto animado) que se parece a una pelota.
Una bola es un sprite redondo que está contenido en un lienzo, puede reaccionar a toques, arrastres, interactuar con otros sprites (sprites de imagen y otras bolas) y con el borde de la tela, y moverse de acuerdo a sus propiedades. Por ejemplo, para que una bola se mueva 4 píxeles en la parte superior de un lienzo cada 500 milisegundos (medio segundo), se puede establecer la propiedad Speed a 4, la propiedad Interval a 500, la propiedad Heading a 90 (grados), y establecer la propiedad Enabled a verdadero. Puede cambiar éstas y otras propiedades para modificar el comportamiento de la bola.
La diferencia entre una bola y un sprite de imagen es que este último puede obtener su apariencia de una imagen, mientras que el aspecto de una bola sólo puede ser cambiado variando sus propiedades PaintColor y Radius.

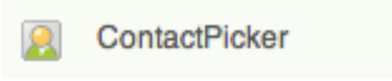
3.2. ImageSprite (Sprite de imagen).



Un componente sprite de imagen es un objeto animado que puede interactuar con un lienzo, las bolas, y otros objetos de imagen.
Un sprite de imagen es un objeto animado que está contenido en un lienzo, puede reaccionar a toques, arrastres, interactuar con otros sprites (sprites de imagen y otras bolas) y con el borde de la tela, y moverse de acuerdo a sus propiedades. Por ejemplo, para tener un movimiento de 10 píxeles a la izquierda cada segundo, se puede establecer la propiedad Speed a 10, la propiedad Interval a 1000 (milisegundos), la propiedad Heading a 180 (grados), y establecer la propiedad Enabled a verdadero. Puede cambiar éstas y otras propiedades para modificar el comportamiento del sprite imagen.

4. Componentes Sociales (Social)

4.1. ContactPicker (Selector de contactos)

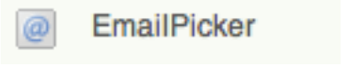


Utilizar un componente selector de contactos para que el usuario seleccione una entrada de la lista de contactos de Android. Un selector de contactos es un botón que muestra una lista de contactos para elegir cuando el usuario lo pulsa. Después de que el usuario ha hecho una selección, quedan establecidas las siguientes propiedades:

• ContactName: el nombre del contacto seleccionado.
• EmailAddress: Dirección de correo electrónico principal del contacto.
• Picture: El nombre del archivo que contiene la imagen del contacto, que puede ser utilizado como imagen en otros componentes tales como Image oImageSprite.

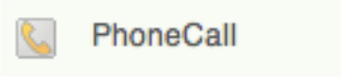
Otras propiedades afectan a la apariencia del botón (incluyendo TextAlignment y BackgroundColor) y si puede ser utilizado (Enabled).

4.2. ContactPicker (Selector de contactos)



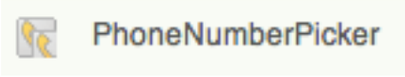
Se puede utilizar un componente selector de correo electrónico para que el usuario introduzca la dirección de un usuario de correo electrónico de la lista de contactos del dispositivo Android.
Un selector de correo electrónico es un cuadro de texto en el que un usuario puede empezar a introducir una dirección de correo electrónico de un contacto que el dispositivo se ofrecerá a autocompletar. El valor inicial del cuadro de texto y el valor tras la entrada del usuario se encuentran en la propiedad Text. Si la propiedad Text está vacía inicialmente, el contenido de la propiedad Hint se mostrará en gris claro como una sugerencia para el usuario.
Otras propiedades afectan a la apariencia del selector de correo electrónico (incluyendo TextAlignment y BackgroundColor) y si se puede utilizar (Enabled).
El selector de correo electrónico se utiliza generalmente con un botón. El usuario pulsa el botón cuando la entrada textual se ha completado.

4.3. PhoneCall (llamada telefónica)



Utilice este componente para marcar el teléfono y hacer una llamada.
PhoneCall es un componente no visible que hace una llamada telefónica al número especificado en la propiedad PhoneNumber, que se puede establecer en el Diseñador o en el Editor de bloques. Se puede utilizar el método MakePhoneCallpara hacer una llamada telefónica mediante la programación de la aplicación.
Este componente se utiliza a menudo con el componente ContactPicker, que permite al usuario seleccionar uno de los contactos almacenados en el teléfono y se establece la propiedad PhoneNumber para el número de teléfono del contacto.
Para especificar directamente el número de teléfono, establezca la propiedad PhoneNumber a un texto que contenga los dígitos especificados (por ejemplo, "6505551212"). El número puede ser formateado con guiones, puntos y paréntesis, sino se ignoran. No se pueden incluir espacios en el número.

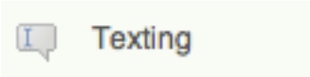
4.4. PhoneNumberPicker (Selector de número de teléfono).



Este componente se utiliza para permitir a los usuarios elegir un número de teléfono de una lista de los números de teléfono de los contactos hallados en el dispositino Android.
Cuando el usuario pulsa un botón selector de número de teléfono, se muestra una lista de los números telefónicos almacenados en los contactos para elegir. Después de que el usuario ha hecho una selección, las siguientes propiedades se ajustarán a la información sobre el contacto:

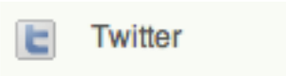
• ContactName: el nombre del contacto.
• PhoneNumbe: número seleccionado.
• EmailAddress : Dirección de correo electrónico principal de contacto.
• Picture: El nombre del archivo que contiene la imagen del contacto, que puede ser utilizado como Picture para el valor de la propiedad Image o del componente ImageSprite.
Otras propiedades afectan a la apariencia del botón (incluyendo TextAlignment y BackgroundColor) y si puede ser utilizado (Enabled).

4.5. Texting (mensajes de texto)



Este componente permite a los usuarios enviar y recibir mensajes de texto.
Cuando se llama al método SendMessage, el componente de mensajes de texto envía el mensaje de texto que se especifica en la propiedad Message al número de teléfono especificado en la propiedad PhoneNumber. El componente de mensajes de texto puede recibir mensajes de texto a menos que la propiedad ReceivingEnabled sea False. Cuando llega un mensaje, se produce el evemto MessageReceived que proporciona el número de teléfono del remitente y el mensaje.
Este componente se utiliza a menudo con el componente ContactPicker, que permite al usuario seleccionar uno de los contactos almacenados en el teléfono y se establece la propiedad PhoneNumber con el número de teléfono del destinatario.
Para especificar directamente el número de teléfono, se puede establecer la propiedad PhoneNumber con los dígitos necesarios (por ejemplo, "6505551212"). El número puede ser formateado con guiones, puntos y paréntesis, que se ignoran. Sin embargo, no se pueden incluir espacios en el número.

4.6. Twitter

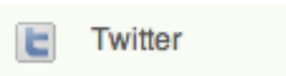


Este componente permite a los usuarios interactuar con Twitter.
Este componente no visible permite la comunicación con Twitter. Los métodos incluidos permiten efectuar búsquedas (SearchTwitter) y acceder a Twitter (Authorize). Una vez que un usuario ha iniciado la sesión y dicho inicio ha sido confirmado con éxito por el evento IsAuthorized, se pueden utilizar los siguientes métodos:

- Establecer el estado del usuario conectado (SetStatus).
- Enviar un mensaje directo a un usuario específico (DirectMessage).
- Recibir los mensajes directos más recientes (RequestDirectMessages).
- Sigue a un usuario (Follow).
- Dejar de seguir un usuario (StopFollowing .)
- Obtener la lista de usuarios que siguen a un usuario (RequestFollowers).
- Obtener los mensajes más recientes de los usuarios seguidos (RequestFriendTimeline).
- Obtener las menciones más recientes del usuario conectado (RequestMentions).

En general, se llama a uno de estos métodos y el resultado estará disponible cuando el correspondiente evento de recepción se dispare. Por ejemplo, si usted llama al método RequestFollowers , entonces el evento FollowersReceived se disparará cuando la lista de seguidores está disponible. Esto puede tardar un rato, porque la solicitud va a través de Internet y el resultado puede incluso no llegar a estar disponible nunca, por ejemplo, cuando el dispositivo no está conectado a la red o el sitio web de Twitter se halle caído.

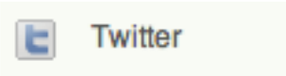
4.6. Twitter



El inicio de sesión se maneja mediante el protocolo OAuth, que es requerido por la API de Twitter (dev.twitter.com/pages/auth) . Una aplicación de App Inventor que necesita hacer llamadas a la API de Twitter en nombre de un usuario autenticado, primero debe llamar al método Authorize . Esto provoca que se presente una página de inicio de sesión de Twitter, que permitirá al usuario ingresar su nombre de usuario y contraseña, en caso necesario, y proporciona las credenciales adecuadas para el componente Twitter de la aplicación. Una vez que el componente Twitter tiene las credenciales el evento IsAuthorized se dispara de modo que se informe a la aplicación que puede proceder a efectuar llamadas a la API de Twitter. El único método del componente Twitter que no requiere que el usuario se acredite es SearchTwitter . Una aplicación de Twitter conserva las credenciales a través de las invocaciones, así que puedeno ser necesario que el usuario inicie sesión cada vez que utilice la aplicación. Reinstalar la aplicación borrará las credenciales, de igual modo que ocurre al invocar el método DeAuthorize. Un usuario también puede revocar la autorización para una aplicación desde el sitio web de Twitter, a través de la página de configuración. Para comprobar si una aplicación ya cuenta con credenciales válidas, utilice el método CheckAuthorized.
Importante: La mayor parte del funcionamiento del protocolo OAuth queda oculto en el interior del componente Twitter y no es necesario entender cómo funciona para poder utilizarlo. Sin embargo cualquiera que necesite construir una aplicación con el componente Twitter deberá suministrar una Consumer Key y una Consumer Secret como propiedades del componente. Ambas son unas cadenas de texto específicas para la aplicación que pueden obtenerse mediante el registro de la aplicación en Twitter en twitter.com/oauth_clients/new . En la página de registro se deberá proporcionar la siguiente información:

Icono de la aplicación
Se puede subir un archivo JPG, GIF o PNG, que se utilizará como icono de la aplicación.
Nombre de la aplicación
Un nombre único para su aplicación. Cuando usted envía el formulario se le avisará si el nombre ya está siendo utilizado. Este nombre se le muestra al usuario de su aplicación cuando se solicita acceso a Twitter como resultado de una llamada al método Authorize.
Descripción
Una descripción de su aplicación.

4.6. Twitter



Sitio web de la aplicación
El sitio web donde el usuario puede obtener más información sobre su aplicación o descargarla, si es que existe.
Organización
Su empresa u organización
Sitio web de la organización
El sitio web de la empresa u organización, u otro sitio web, ya que este campo no puede dejarse en blanco.
Tipo de aplicación
Este campo debe establecerse en browser (navegador).
URL de vuelta
Una dirección URL válida. El valor no tiene importancia ya que el componente de Twitter establece el valor adecuado.
Tipo de acceso por defecto
Establecer en Read & Write (lectura / escritura)

Los demás campos se pueden dejar en blanco si lo desea. Una vez que haya registrado con éxito su aplicación verá una página que muestra las cadenas de texto correspondientes a la Consumer Key y Consumer Secret específicas de la aplicación registrada. Puede copiarlas en las propiedades correspondientes del componente Twitter en su aplicación de App Inventor. Si desea cambiar la configuración de su aplicación en un momento posterior, iniciar sesión en Twitter a través de un navegador web e ir a twitter.com/apps para ver y modificar las aplicaciones registradas.

5. Componentes Sensores (Sensor).

5.1. AccelerometerSensor (Sensor acelerómetro).

Este componente recibe señales del acelerómetro del dispositivo Android, que detecta sacudidas y mide la aceleración en tres dimensiones. La aceleración se mide en unidades del SI (m /s2). Si el dispositivo permanece apoyado en reposo sobre su parte posterior, la aceleración Z será de aproximadamente 9,8 m/s2.
El componente produce tres valores

• XAccel : Positivo cuando el dispositivo se inclina hacia la derecha (es decir, su lado izquierdo se levanta), y negativo cuando el dispositivo se inclina hacia la izquierda (su borde derecho se levanta).

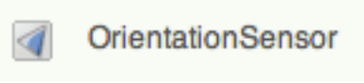
• YAccel: positivo, cuando la parte inferior del dispositivo (la del micrófono) se eleva, y negativa cuando su parte superior (la del auricular) se eleva.

• ZAccel: Positivo cuando la pantalla esté mirando hacia arriba, y negativa cuando la pantalla está mirando hacia abajo.

5.2. LocationSensor (Sensor de posicionamiento).

Este componente proporciona la ubicación del dispositivo Android, utilizando el GPS si está disponible o un método alternativo en caso contrario, como pueden ser las torres de telefonía móvil o las redes inalámbricas conocidas.
LocationSensor es un componente no visible que puede proporcionar información sobre la ubicación, incluyendo longitud, latitud, altitud (si el dispositivo posee esta capacidad), y la dirección. Este componente también puede proporcionar geocodificación, la conversión de una dirección dada (no necesariamente la actual) a una latitud (con el método LatitudeFromAddress) y una longitud (con el método LongitudeFromAddress).
Para funcionar, el componente debe tener la propiedad Enabled establecida, y el dispositivo debe tener habilitada la detección de localización a través de satélites GPS (hallándose con cobertura del sistema) o un método alternativo.

5.3. OrientationSensor (Sensor de orientación).

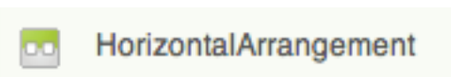


Puede utilizarse el componente sensor de orientación para determinar la orientación espacial del teléfono.
El sensor de orientación es un componente no visible, que informa de los siguientes tres valores, en grados:
• **Giro:** 0 grados, cuando el dispositivo esté nivelado, que aumenta a 90 grados cuando el dispositivo se inclina sobre su lado izquierdo, y disminuye hasta -90 grados cuando el dispositivo se inclina sobre su lado derecho.
• **Inclinación:** 0 grados, cuando el dispositivo esté nivelado, que aumenta a 90 grados cuando el dispositivo se inclina sobre su parte superior (la del auricular), aumentando aún más, hasta 180 grados, cuando se da la media vuelta. Del mismo modo, cuando el dispositivo se inclina sobre su parte inferior (la del micrófono), el ángulo disminuye a -90 grados, llagando a -180 grados, al dar la media vuelta.
• **Azimut:** 0 grados, cuando la parte superior del dispositivo (la del auricular) está apuntando hacia el norte, 90 grados cuando se está apuntando al este, 180 grados cuando se está apuntando al sur, 270 grados cuando se apunta al oeste, etc.

Estas mediciones suponen que el propio dispositivo no está en movimiento.

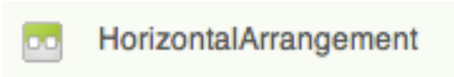
6. Componentes Alineamientos de Pantalla.

6.1. HorizontalArrangement (Alineamiento horizontal).



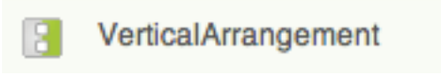
Utilizar un componente de alineamiento horizontal para mostrar un grupo de componentes determinado de izquierda a derecha. Este componente es un elemento de formato que permite colocar los componentes que deben mostrarse de izquierda a derecha. Si, por el contrario, desea tener los componentes uno sobre otro, utilice VerticalArrangement en su lugar.
Con un HorizontalArrangement, los componentes están dispuestos a lo largo del eje horizontal, con un alineamiento vertical centrado.
Si la propiedad Height (altura) de un HorizontalArrangement se halla establecida en automático, su altura será determinada por el componente más alto que contenga cuya propiedad altura no esté establecida para llenar el patrón (su recuadro). Si la propiedad altura se establece en automático y sólo comprende elementos cuya altura está establecida para llenar el patrón, la altura final se calcula utilizando la altura automática de los componentes. Si la propiedad altura de un HorizontalArrangement está establecida en automático y no contiene nada, la altura será de 100.
Si la propiedad anchura de un HorizontalArrangement se establece en Automático, su anchura real estará determinada por la suma de las anchuras de los componentes que contenga. Si la propiedad anchura de un HorizontalArrangement se establece en automático, todos los componentes cuya anchura se halle establecida para llenar el patrón se comportarán como si su anchura hubiera estado establecida en automático.
Si la propiedad anchura de un HorizontalArrangement se establece para llenar el patrón o se especifica en píxeles, cualquier componente cuya anchura esté establecida para llenar el patrón ocupará la anchura no ocupada por otros componentes que compartan el mismo HorizontalArrangement.

6.2. TableArrangement (Alineamiento en tabla).



Se puede utilizar un componente de alineamiento en tabla para mostrar un grupo de componentes de una forma tabular.
Este componente es un elemento de formato que coloca los componentes que deben mostrarse en forma de tabla.
En un TableArrangement, los componentes están dispuestos en una cuadrícula de filas y columnas, con no más de un componente visible en cada celda. Si varios componentes ocupan la misma celda, sólo el último será visible.
Dentro de cada fila, los componentes son alineados verticalmente en posición central.
La anchura de una columna está determinado por el mayor componente de esa columna. Al calcular la anchura de la columna, el ancho automático se utiliza para aquellos componentes cuya propiedad anchura se halla establecida para llenar la celda. Sin embargo, cada componente siempre llenará el ancho total de la columna que ocupa.
La altura de una fila es determinada por el componente más alto en la fila cuya propiedad altura no esté establecida para llenar dicha fila. Si una fila contiene sólo componentes cuya altura se halle establecida para llenar la fila, la altura de la fila se calcula utilizando la altura automática de los componentes.

6.3. VerticalArrangement (Alineamiento vertical)



Se puede utilizar un componente de alineamiento vertical para mostrar un grupo de componentes dispuestos de arriba hacia abajo, alineados a la izquierda de la pantalla.
Este componente es un elemento de formato en el cual pueden colocarse componentes que deban mostrarse uno debajo de otro. El primer componente se muestra en la parte superior, el segundo justo debajo, y así sucesivamente. Si, por el contrario, desea que los componentes se muestren uno junto a otro, utilice un componente HorizontalArrangement en su lugar.
En una VerticalArrangement, los componentes están dispuestos a lo largo del eje vertical, y el propio VerticalArrangement queda alineado a la izquierda.
Si la propiedad Width (anchura) de un VerticalArrangement se halla establecida en automático, su anchura será determinada por el componente más ancho que contenga, cuya propiedad anchura no esté establecida para llenar el patrón (su recuadro). Si la propiedad anchura se establece en automático y sólo comprende elementos cuya anchura está establecida para llenar el patrón, la anchura final se calcula utilizando la anchura automática de los componentes. Si la propiedad anchura de un VerticalArrangement está establecida en automático y no contiene nada, la anchura será de 100.
Si la propiedad altura de un VerticalArrangement se establece en Automático, su altura final estará determinada por la suma de las alturas de los componentes que contenga. Si la propiedad altura de un VerticalArrangement se establece en automático, todos los componentes cuya altura se halle establecida para llenar el patrón se comportarán como si su altura hubiera estado establecida en automático.
Si la propiedad altura de un VerticalArrangement se establece para llenar el patrón o se especifica en píxeles, cualquier componente cuya altura esté establecida para llenar el patrón ocupará la altura no ocupada por otros componentes que compartan el mismo VerticalArrangement.

7. Componentes LEGO MINDSTORMS

Estos componentes proporcionan un control de LEGO MINDSTORMS NXT robots a través de Bluetooth.

7.1. NxtDirectCommands

Un componente que proporciona una interfaz de bajo nivel para un robot LEGO Mindstorms NXT, con funciones para enviar NXT Comandos directos.

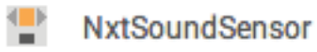
7.2. NxtColorSensor

Un componente que proporciona una interfaz de alto nivel para un sensor de color en un robot LEGO Mindstorms NXT.

7.3. NxtLightSensor

Un componente que proporciona una interfaz de alto nivel para un sensor de luz en un robot LEGO Mindstorms NXT

7.4. NxtSoundSensor



Un componente que proporciona una interfaz de alto nivel para un sensor de sonido en un robot LEGO Mindstorms NXT

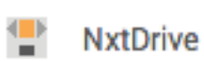
7.5. NxtTouchSensor

Un componente que proporciona una interfaz de alto nivel a un sensor táctil en un robot LEGO Mindstorms NXT

7.6. NxtUltrasonicSensor

Un componente que proporciona una interfaz de alto nivel a un sensor táctil en un robot LEGO Mindstorms NXT

7.7. NxtDrive



Un componente que proporciona una interfaz de alto nivel para un robot LEGO Mindstorms NXT, con las funciones que se pueden mover y girar el robot.

8. Otros Componentes

8.1. ActivityStarter (Arranque de actividades)

Un componente que puede poner en marcha otras actividades (aplicaciones, funciones del teléfono...) desde su aplicación. Usted se comunica con el motor de arranque de la actividad mediante el establecimiento de las propiedades del componente para pasar la información relacionada con la actividad, incluyendo la acción y la clase de actividad. Consulte Uso del componente ActivityStarter para más detalles y ejemplos.
Las actividades que pueden ser lanzadas incluyen:

• Otra aplicación de App Inventor para Android. Para ello, primero se determina la clase de la otra aplicación mediante la descarga del código fuente y el uso de un explorador de archivos o se descomprime la aplicación y se localiza un archivo llamado "youngandroidproject/project.properties". La primera línea del archivo comenzará con "main=" seguido por el nombre de la clase, por ejemplo,
main=com.gmail.Bitdiddle.Ben.HelloPurr.Screen1

Para hacer que ActivityStarter pueda arrancar esta aplicación, establezca las siguientes propiedades:

• ActivityPackage al nombre de clase, eliminando el último componente (por ejemplo, com.gmail.Bitdiddle.Ben.HelloPurr)
• ActivityClass al nombre de clase completo (por ejemplo, com.gmail.Bitdiddle.Ben.HelloPurr.Screen1)
• Una funcionalidad integrada en el sistema operativo Android, como el uso de la cámara, o la ejecución de una búsqueda en la Web. Usted puede iniciar la cámara mediante el establecimiento de las siguientes propiedades:
• Action: android.intent.action.MAIN
• ActivityPackage: com.android.camera
• ActivityClass: com.android.camera.Camera

8.1. ActivityStarter (Arranque de actividades)

• Realizar una búsqueda en la Web: Suponiendo que el término que desea buscar es "vampiro" (no dude en sustituir por su propia elección), establezca las propiedades de este modo:

• Action: android.intent.action.WEB_SEARCH
• ExtraKey: query
• ExtraValue: vampiro
• ActivityPackage: com.google.android.providers.enhancedgooglesearch
• ActivityClass: com.google.android.providers.enhancedgooglesearch.Launcher
• La presentación en el navegador de una página web especificada. Suponiendo que la página que desea es ir a "www.facebook.com" (no dude en sustituirla por su propia elección), establezca las propiedades como sigue:

• Action: android.intent.action.VIEW
• DataUri: http://www.facebook.com

También puede ejecutar aplicaciones de terceros instaladas en el teléfono, siempre y cuando conozca las propiedades adecuadas para invocarla, y también se pueden poner en marcha actividades que producen resultados de texto y reutilizarlos en su aplicación. La forma de extraer esos datos depende de cómo se haya implementado la aplicación.

8.2. BarcodeScanner (Lector de código de barras)

Este componente no visible utiliza la cámara del teléfono para leer un código de barras unidimensional o bidimensional (código QR). Para que este componente funcione, la aplicación de escáner de código de barras de ZXing debe estar instalada en el teléfono. Esta aplicación está disponible de forma gratuita en el Android Market.

8.3. BluetoothClient (Cliente de Bluetooth)

BluetoothClient

8.4. BluetoothServer (Servidor de Bluetooth)

BluetoothServer

Componente servidor de Bluetooth.

8.5. Notifier (Notificador)

Notifier

El notificador es un componente no visible que puede mostrar varios tipos de alertas y puede registrar las respuestas del usuario a las mismas. Use un notificador para mostrar avisos y alertas a los usuarios de su aplicación, y también para registrar la información que puede ayudarle a depurar su aplicación.

8.6. SpeechRecognizer (Reconocedor de voz)

Este componente no visible utiliza la cámara del teléfono para leer un código de barras unidimensional o bidimensional (código QR). Para que este componente funcione, la aplicación de escáner de código de barras de ZXing debe estar instalada en el teléfono. Esta aplicación está disponible de forma gratuita en el Android Market.

8.7. TextToSpeech (Sintetizador de voz)

Utilizar un componente sintetizador de voz para que el dispositivo lea un texto de modo audible.
El componente sintetizador de voz tiene propiedades que se pueden establecer para guiar la pronunciación del texto articulado. Estas propiedades utilizan códigos de tres letras para especificar el idioma y el país en el que se habla esa lengua. Por ejemplo, puede especificar Inglés Británico o Inglés EE.UU.. Para Inglés Británico, el código de idioma es eng y el código del país es GBR , mientras que para el inglés de EEUU, el código de idioma es eng y el código del país es USA . La lista completa está a continuación.

8.8. TinyWebDB (base de datos web)

Componente no visible que se comunica con un servicio Web para almacenar y recuperar información.
Consulte Creación de un servicio personalizado de TinyWebDB.

8.9. Web

Componente no visible componente que proporciona funciones de HTTP GET y POST.

9. Webgrafía

https://sites.google.com/site/aprendeappinventor/