

DIXITALIZACIÓN

Curso: 4º ESO

APP inventor

Guía turística



Visor, componentes y medios de la app
Guía turística.

Guía turística es una *app* que muestra información de interés para visitar una localidad.

La pantalla principal es un menú compuesto por una imagen de fondo y varios botones. Cada botón abrirá una nueva pantalla con la información correspondiente; por ejemplo, «Lugares de interés»:

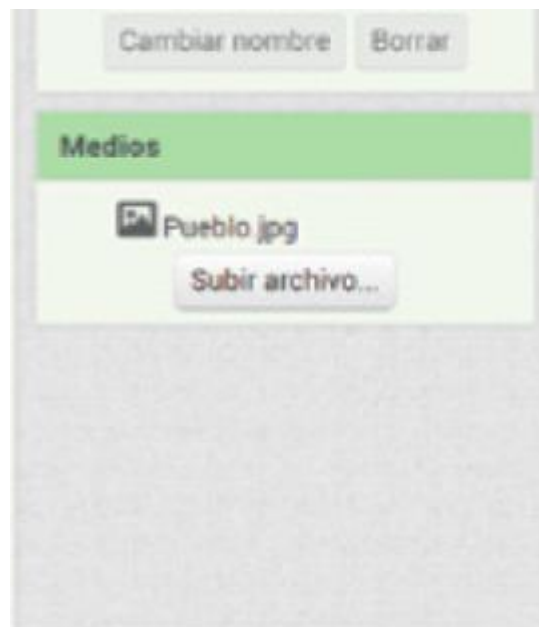
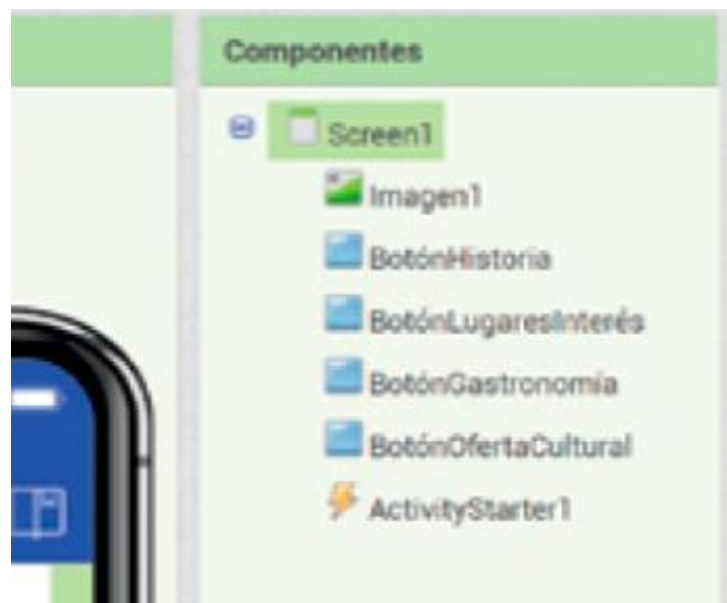


Desde la barra de herramientas superior se pueden añadir las ventanas deseadas y asignarles nombre. Dentro de cada pantalla será necesario colocar un botón para regresar al menú:



El botón «Oferta cultural» abre la página oficial de la localidad, utilizando el elemento **ActivityStarter**, tal y como se muestra:





Visor, componentes y medios de la app
Guía turística.

Mascota virtual



Visor, componentes y medios de la app Mascota Virtual.

Diseño de la *app*

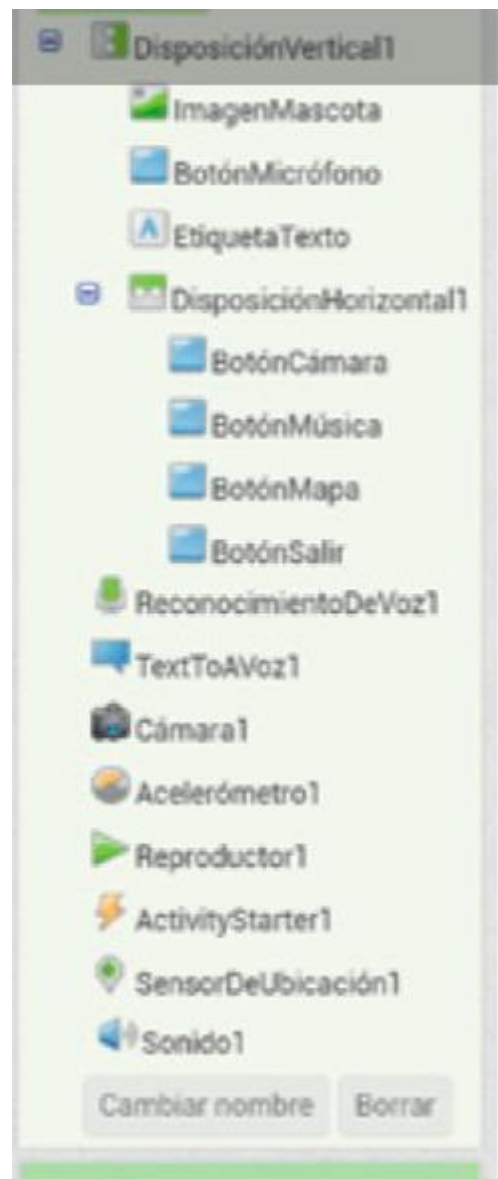
Para distribuir los objetos en la pantalla, se utiliza una disposición vertical, que incluye una imagen para la mascota, un botón para el micrófono, una etiqueta para mostrar mensajes y una disposición horizontal para alinear los botones (foto, música, mapa y salir).

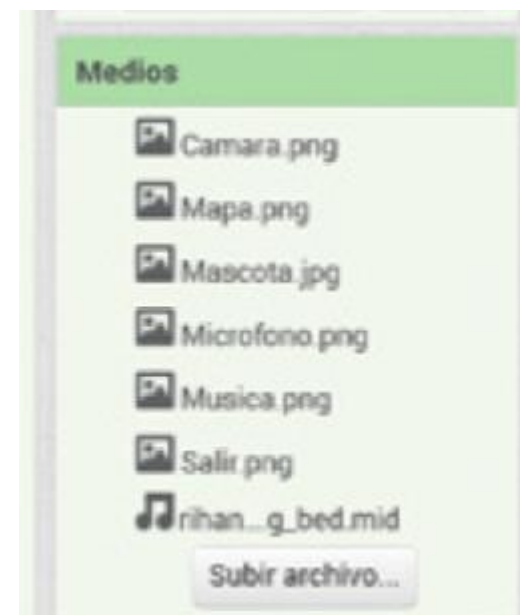
Las propiedades de cada botón se han ajustado para personalizar la imagen que muestran, eliminar el texto que aparece por defecto, así como ajustar los parámetros alto y ancho.

Para desarrollar la funcionalidad de la aplicación, hay que añadir los siguientes componentes no visibles: **Cámara, Reproductor, Sonido, SensorDeUbicación, ActivityStarter, TextoAVoz, Acelerómetro y ReconocimientoDeVoz.**



Visor, componentes y medios de la app Mascota Virtual.





Visor, componentes y medios de la app Mascota Virtual.

Abrir y cerrar la aplicación

El bloque **Screen1.Inicializar** se ejecuta al inicio de la aplicación y la mascota comienza saludando con el mensaje «¡Hola!». Finalmente, se utiliza el evento **cuando BotónSalir.Clic** para cerrar la aplicación y que no se quede residente en memoria. Esta funcionalidad solamente se puede probar en la aplicación descargada, ya que durante el desarrollo está deshabilitada.

Abrir la cámara y tomar foto

La cámara del dispositivo móvil se pone en funcionamiento cuando se detecta el evento **BotónCámara.Clic**, que llama al procedimiento **AbrirCámara** y realiza una fotografía utilizando el método **Cámara1. TomarFoto**

cuando Screen1 .Inicializar
ejecutar llamar TextToAVoz1 .Hablar
mensaje ¡Hola!

cuando BotónSalir .Clic
ejecutar cerrar la aplicación

cuando BotónCámara .Clic
ejecutar Llamar AbrirCámara

como AbrirCámara
ejecutar llamar Cámara1 .TomarFoto

cuando BotónMúsica .Clic
ejecutar Llamar SonarMúsica

como SonarMúsica
ejecutar si Reproductor1 .Sonando
entonces llamar Reproductor1 .Pausar
sino llamar Reproductor1 .Iniciar

cuando Acelerómetro1 .Agitar
ejecutar llamar Sonido1 .Vibrar
milisegundos 2000
llamar Sonido1 .Reproducir
llamar TextToAVoz1 .Hablar
mensaje ¡Me haces vibrar!

Bloques de la app Mascota virtual.

Reproducir música

El procedimiento **SonarMúsica** comprueba si el reproductor de música está sonando; si es así, **Reproductor1.Pausar** para la música y, si no es así, **Reproductor1.Iniciar** la activa. La canción, en formato MP3, se añade utilizando la propiedad **Origen**, ya sea en el diseñador o en el editor de bloques.

Usar el acelerómetro

El acelerómetro del dispositivo móvil detecta cuando se está agitando, con el evento **Acelerómetro1.Agitar**, y ejecuta tres llamadas: **Sonido1.Vibrar**, que activa la vibración del dispositivo durante los milisegundos especificados; **Sonido1.Reproducir**, que hace sonreír a la mascota al reproducir el archivo «Sonrisa.mp3» asociado a **Sonido1** y **TextToAVoz1.Hablar**, que pronuncia el mensaje «¡Me haces vibrar!».

Abrir en el navegador

El componente **ActivityStarter** puede poner en marcha diversas aplicaciones y funciones del teléfono. En este caso, se utiliza para abrir el navegador utilizando una dirección en Google Maps.



Reconocimiento de voz

El componente **ReconocimientoDeVoz** convierte el lenguaje hablado en texto usando la función de reconocimiento de voz de Android. En primer lugar, hay que utilizar **ObtenerTexto** para activar el micrófono mientras la persona usuaria habla. A continuación, una vez que la voz se ha convertido en texto, se lanza el evento **DespuésDeObtenerTexto** que contiene el resultado.



En la aplicación, el texto obtenido se compara con cada una de las instrucciones que puede decir la persona usuaria: «haz una foto», «pon música» y «abre el mapa», para lanzar el procedimiento correspondiente. En caso de que el texto reconocido no corresponda con ninguno de los anteriores, la mascota lanza un mensaje con instrucciones de ayuda.

cuando BotónMicrófono .Clic
ejecutar llamar ReconocimientoDeVoz1 .ObtenerTexto

cuando ReconocimientoDeVoz1 .DespuésDeObtenerTexto
Resultado partial
ejecutar poner EtiquetaTexto . Texto como tomar Resultado
si
call comparar
instrucción tomar Resultado
acción " haz una foto "
entonces Llamar AbrirCámara
si no, si
call comparar
instrucción tomar Resultado
acción " pon música "
entonces Llamar SonarMúsica
si no, si
call comparar
instrucción tomar Resultado
acción " pon música "
entonces Llamar SonarMúsica
si no, si
call comparar
instrucción tomar Resultado
acción " abre el mapa "
entonces Llamar AbrirNavegador
sino
llamar TextToAVoz1 .Hablar
mensaje " No te he e