

GUÍA ACTIVIDAD

RESULTADOS INTELECTUALES

ESCUELAS INNOVADORAS ADAPTADAS A LA SOCIEDAD DIGITAL
PARA MEJORAR LAS COMPETENCIAS EDUCATIVAS TECNOLÓGICAS

Proyecto nº 2020-1-ES01-KA201-082648



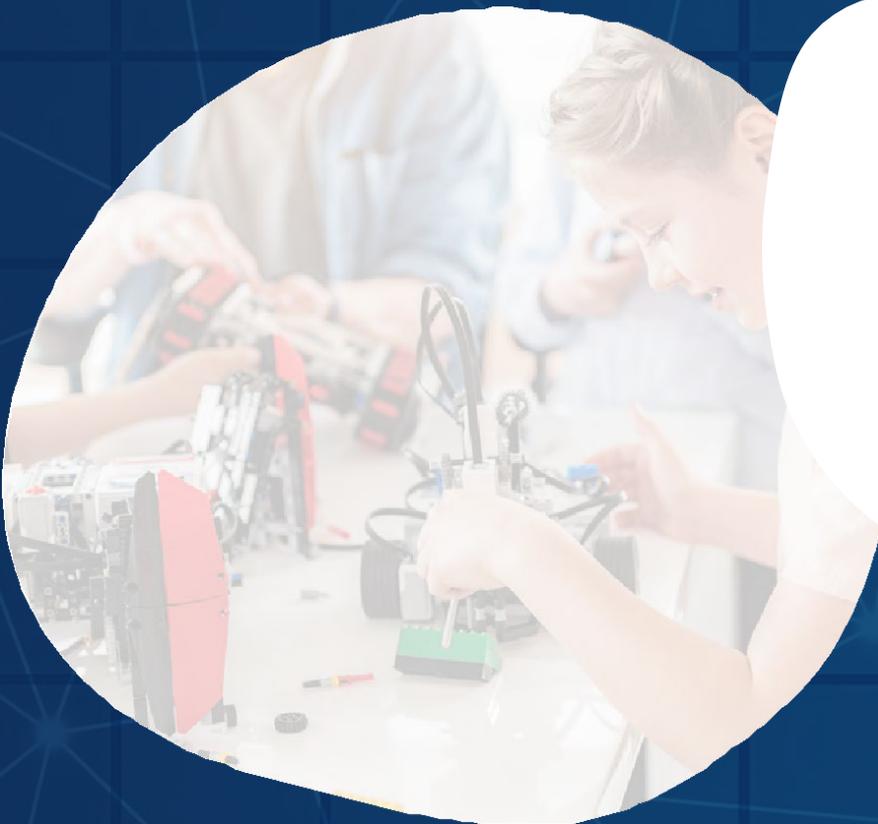
Co-funded by the
Erasmus+ Programme
of the European Union



PRODUCTO INTELECTUAL 1

ROBÓTICA EDUCATIVA

LA ROBÓTICA EDUCATIVA ES UN MÉTODO INTERDISCIPLINAR EN EL QUE SE TRABAJAN MATERIAS COMO LAS MATEMÁTICAS, LA TECNOLOGÍA, LA CIENCIA Y LA INGENIERÍA. GRACIAS A ELLA, LOS NIÑOS PUEDEN DESARROLLAR EL PENSAMIENTO LÓGICO, LA IMAGINACIÓN Y LA LINGÜÍSTICA



1

ACTIVIDAD 1

TÍTULO BÚSQUEDA DEL TESORO PIRAMIDAL

RESUMEN

Inspirados por la exploración de la cámara oculta de la Gran Pirámide de Giza, monumento egipcio de 4.500 años de antigüedad, por robots y la necesidad de explorar espacios de imposible acceso para los humanos, intentamos simular una situación similar en clase. La actividad supuso un gran reto: los alumnos de 7º D propusieron toda una lista de ideas sobre cómo podríamos organizar el túnel, cómo controlar el robot en el túnel, cómo filmar el recorrido y cómo visualizarlo. La actividad resultó muy atractiva y la participación de los alumnos fue excelente.

El objetivo de esta actividad es que queremos enseñar a nuestros alumnos 2 cosas básicas en nuestro proyecto llamado "Pyramid Treasure Hunt".

1. Utilizar el kit del robot Arduino 4WD,
 2. Utilizar el software Appinventor,
- Haremos estos estudios a nivel de enseñanza secundaria. Utilizaremos como software programas basados tanto en texto como en bloques. También produciremos grandes obras con una impresora 3D.

AUTOR/S

Instituto Maria Rosetti

FECHA 06/12/2022

VERSIÓN 1

OBJETIVOS DIDÁCTICOS

1. El alumno sabe utilizar el software "Thingiverse".
2. En el software Student Appinventor;
 - 2.1 Saber cómo crear una cuenta
 - 2.2 Saber utilizar la pantalla de diseño de Appinventor
 - 2.3 Saber utilizar la pantalla de codificación de Appinventor.

ACTIVIDAD 1

3. En el software Arduino Ide para estudiantes;

3.1 Saber instalar el programa

3.2 Saber definir y utilizar variables para contener datos en el programa. 3.3

Ordenar la entrada y salida de los pines de Arduino.

3.4 Crear el algoritmo del programa y escribir el código.

4. Montar el robot;

4.1 Arduino se conecta a los pines correctos en la placa

4.2 Establecer la conexión entre la placa Arduino y el controlador del motor.

4.3 Establecer la conexión entre el conductor del motor y la moto.

4.4 Establecer la conexión entre el módulo Bluetooth y el software Android.

CIENCIA

IDIOMAS

TECNOLOGÍA

LITERATURA

MATEMÁTICAS

MÚSICA

GEOGRAFÍA/HISTORIA

OTROS

NIVEL DE ESTUDIOS

Esta actividad está preparada para...

12 - 14 AÑOS

14 - 16 AÑOS

OTROS

HERRAMIENTAS NECESARIAS

-Contenido del kit Arduino 4WD

- 4 piezas - ruedas de 65 mm de diámetro.

- 4 piezas - Motorreductor de plástico + disco codificador.

- 2 Piezas - Parte superior e inferior de la carrocería de plexiglás.

- Cama de batería.

- Accesorios mecánicos y electrónicos.

-Herramientas necesarias para soldar (se debe buscar ayuda de un adulto al soldar).

-Impresora 3D

-Software (Arduino Ide, Appinventor, Thingiverse)

-Teléfono móvil o cámara

ACTIVIDAD 1

DESARROLLO DE LA ACTIVIDAD

Primer paso: Diseñar con el software Thinkcard

El objetivo de nuestro proyecto era buscar tesoros fabricando un robot que podamos mover en la dirección que queramos. Por esta razón, hemos diseñado objetos que podemos considerar tesoros. De nuevo, como el proceso de búsqueda tendrá lugar en Egipto según el proyecto, también podemos diseñar objetos específicos de esa zona. Por ejemplo, la Pirámide. Para ello, necesitamos programas de diseño 3D.

Segundo Paso: Desarrollo de aplicaciones móviles con Appinventor(Panel de diseño)

Utilizamos los botones para determinar las direcciones. Escribimos "Delante", "Izquierda", "Derecha", "Atrás" en el campo de texto de los botones. Para parar, creamos un botón "Stop". Añadimos dos botones donde podemos ver el estado de la conexión. Puedes personalizar el diseño a tu gusto. Colores de los botones, fuentes, algunas etiquetas extra, etc. Nosotros elegimos un diseño sencillo. Además de estos, vamos a añadir los elementos "Cliente Bluetooth" y "Servidor Bluetooth". Pero estos son los componentes invisibles.

Tercer Paso: Desarrollo de aplicaciones móviles con Appinventor(Code Panel)

Para cada botón, debemos escribir los códigos que se ejecutarán cuando "Toque abajo" y "Toque arriba". También determinamos aquí los valores numéricos que utilizaremos en el entorno Arduino Ide.

A continuación, tenemos que escribir los códigos para conectarse a Bluetooth.

Cuarto Paso: Software Arduino Ide

Códigos Arduino

[c]

```
int pwm1 = 10;
```

```
int pwm2 = 11;
```

```
int way1 = 12;
```

```
int way2 = 13;
```

ACTIVIDAD 1

```
int received_data = 0;
void setup()
{
  pinMode(pwm1, OUTPUT);
  pinMode(pwm2, OUTPUT);
  pinMode(yon1, OUTPUT);
  pinMode(yon2, OUTPUT);
  digitalWrite(pwm1, LOW);
  digitalWrite(pwm2, LOW);
  digitalWrite(yon1, LOW);
  digitalWrite(yon2, LOW);
  Serial.begin(9600);
}
void MotorControl(int mway1, int mway2, int pwmInput)
{
  digitalWrite(yon1, mway1);
  digitalWrite(yon2, mway2);
  digitalWrite(pwm1, pwmInput);
  digitalWrite(pwm2, pwmInput);
}
void loop()
{
  if (Serial.available() > 0)
  {
    received_data = Serial.read();
    if (received_data == 10) //Front
    {
      MotorControl(BAJO, BAJO, ALTO);
    }
    else if (received_data == 20) // Back
    {
      MotorControl(HIGH, HIGH, HIGH);
    }
    else if (received_data == 30) // Left
    {
      MotorControl(HIGH, LOW, HIGH);
    }
  }
}
```

ACTIVIDAD 1

```
else if (received_data == 40) //Right
{
MotorControl(LOW, HIGH, HIGH);
}
else // If Receive another data, stop
{
MotorControl(LOW, LOW, LOW);
}
}
}
[/c]
```

RECURSOS



ACTIVIDAD 1

```
when Connect -> BeforePicking
do set Connect -> Elements to BluetoothClient1 -> AddressesAndNames

when Connect -> AfterPicking
do / call BluetoothClient1 -> Connect
address Connect -> Selection
then set Connection_status -> Text to "Connected"

when Left -> TouchUp
do call BluetoothServer1 -> SendByteNumber
number "50"

when Left -> TouchDown
do call BluetoothServer1 -> SendByteNumber
number "30"

when Front -> TouchUp
do call BluetoothServer1 -> SendByteNumber
number "50"

when Front -> TouchDown
do call BluetoothServer1 -> SendByteNumber
number "40"

when Right -> TouchUp
do call BluetoothServer1 -> SendByteNumber
number "50"

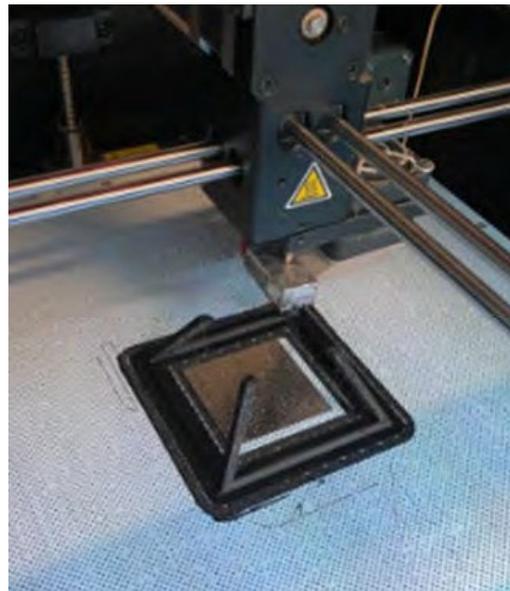
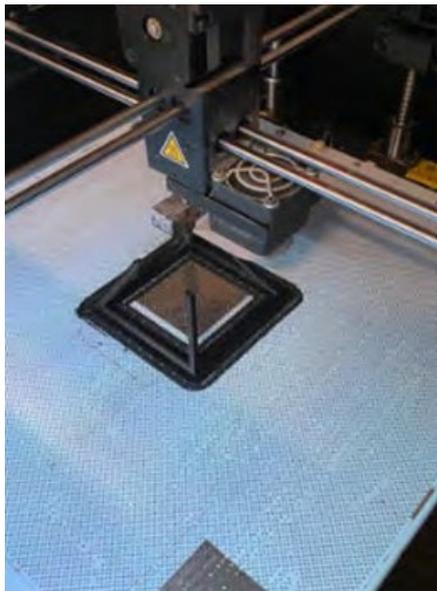
when Right -> TouchDown
do call BluetoothServer1 -> SendByteNumber
number "40"

when Back -> TouchUp
do call BluetoothServer1 -> SendByteNumber
number "50"

when Back -> TouchDown
do call BluetoothServer1 -> SendByteNumber
number "40"
```



ACTIVIDAD 1



ACTIVIDAD 1



EVALUACIÓN DEL ALUMNO

1. ¿Puede el alumno montar correctamente el robot desmontado?
2. ¿Cumple el alumno las normas de seguridad durante la asamblea?
3. ¿Puede el alumno diseñar en la plataforma 3D?
4. ¿Puede el alumno imprimir desde una impresora 3D?
5. ¿Puede el alumno escribir código en el entorno Arduino Ide? ¿Puede el estudiante escribir código en el entorno Appinventor?
7. ¿Puede el alumno establecer una conexión entre el robot y el dispositivo móvil?
8. ¿Puede el alumno colocar correctamente la cámara en el robot?
9. ¿Puede el alumno encontrar la Pirámide y los Tesoros que busca después de ver las imágenes?
10. ¿Pueden trabajar en armonía con el grupo de estudiantes?

BIBLIOGRAFÍA

- <https://www.thingiverse.com/>
- <https://appinventor.mit.edu/>
- <https://www.arduino.cc/>

ADAPTACIÓN A OTROS NIVELES

El proyecto está hecho actualmente para 4 direcciones. En fases posteriores podrán añadirse direcciones intermedias. El objetivo actual del estudio es encontrar un tesoro observando la grabación de la cámara del robot.

MÁS INFORMACIÓN

Esta situación puede mejorarse y la búsqueda del tesoro puede ser más rápida y sencilla gracias a la transferencia de instantáneas.

ACTIVIDAD 2

TÍTULO Pruebas tecnológicas

RESUMEN

Nuestros alumnos se divertieron mucho con la tecnología utilizando algunos objetos de robótica:

- Programaron el robot Boost utilizando la aplicación específica, creando los movimientos correctos para cubrir el itinerario..;
- Ensamblaron correctamente los circuitos eléctricos de Little Bits para hacer que los objetos se movieran;
- Experimentaron las gafas tridimensionales eligiendo juegos entre los propuestos por la lente 3D.

AUTOR/S

IPS Maffeo Pantaleoni

FECH 28/02/2022

VERSIÓN 1

OBJETIVOS DIDÁCTICOS

Los objetivos que deben alcanzarse son los siguientes

- fomentar el aprendizaje interdisciplinar y el trabajo en equipo
- mejorar los intereses y aumentar la motivación
- estimular la creatividad
- mejorar la capacidad de resolución de problemas
- estimular el conocimiento de la electrónica

CIENCIA

TECNOLOGÍA

MATEMÁTICAS

GEOGRAFÍA/HISTORIA

IDIOMAS

LITERATURA

MÚSICA

OTROS

ACTIVIDAD 2

NIVEL DE ESTUDIOS

Esta actividad está preparada para...

12 - 14 AÑOS

14 - 16 AÑOS

OTROS

HERRAMIENTAS NECESARIAS

- Lego Boost
- Gafas 3D
- Objetos diversos: papel, tijeras, plástico, cinta adhesiva, pegamento, limpiapipas...
- catálogo de los modelos a realizar

DESARROLLO DE LA ACTIVIDAD

Los alumnos utilizaron los objetos de las siguientes maneras:

- ordenaron el cartón del itinerario
- pasearon el robot
- programaron los movimientos del robot con Lego Boost App.

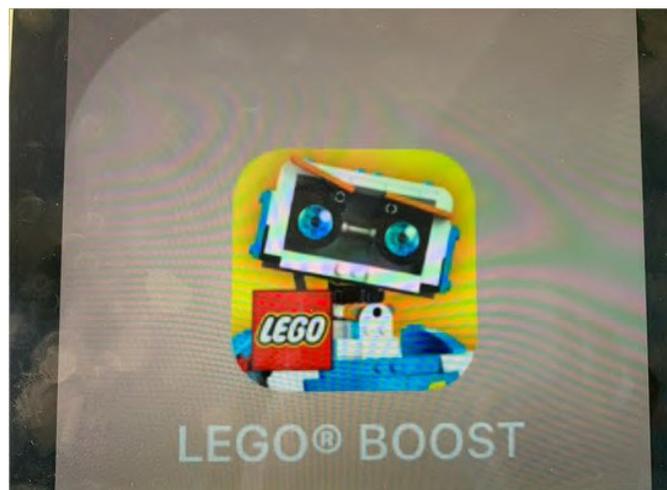
- probaron los distintos movimientos hasta encontrar los correctos que permitieran al robot llegar al final de su recorrido.
- organizaron los bloques eléctricos
- ensamblaron los bloques para oír sonidos, encender y apagar las luces, pegar los distintos bloques a los objetos para que se movieran.
- se ponían las gafas interactivas 3D y realizaban los movimientos propuestos por el juego elegido.

ACTIVIDAD 2

RECURSOS



ACTIVIDAD 2



ACTIVIDAD 2



ACTIVIDAD 2

EVALUACIÓN DEL ALUMNO

Evaluamos a los alumnos en función de la observación del logro de las competencias requeridas por la actividad.

BIBLIOGRAFÍA

Catálogo Little Bits

ADAPTACIÓN A OTROS NIVELES

Para adquirir las destrezas básicas de codificación, los alumnos más jóvenes podrían utilizar bloques digitales y físicos más sencillos.

MÁS INFORMACIÓN

Se invitará a los alumnos a utilizar numerosas herramientas complejas de robótica y objetos tecnológicos físicos y digitales como Arduino.

ACTIVIDAD 3

TÍTULO TRADUCTOR Istedu

RESUMEN

Diseño de una aplicación para teléfonos móviles utilizando App Inventor. La aplicación incorpora un sistema de reconocimiento de voz que captura lo que se habla en forma de texto. Este texto se puede traducir a distintos idiomas y, según la configuración del móvil, se puede escuchar el resultado de la traducción.

Es muy importante tener en cuenta que esta aplicación sólo puede ejecutarse en teléfonos móviles con sistema Android.

AUTOR/S

FECHA 04/04/2022

VERSIÓN 1

IES MEDITERRÁNEO

OBJETIVOS DIDÁCTICOS

Profundizar en el desarrollo de la programación con pseudocódigo. Trabajaremos aspectos relacionados con:

- Pensamiento computacional
- Planificar una aplicación móvil
- Conocer los componentes básicos de una App
- Diseño de las pantallas y colocación de los componentes de la aplicación
- Programación de los componentes de la aplicación

Estrategia de programación: Se utilizará una estrategia basada en el Pensamiento Computacional. Dividiremos el problema global, la creación de una aplicación para smartphones que implemente un traductor, en problemas más sencillos relacionados con cada uno de los componentes que la integran. Finalmente, se integran las soluciones en un proyecto global y se comprueban los posibles errores que puedan surgir para ofrecer nuevas soluciones o mejoras a la aplicación.

ACTIVIDAD 3

El desarrollo de la aplicación consta de dos fases:

- Fase de diseño:

Se diseñan las pantallas y la correcta colocación de los componentes que conforman la aplicación.

- Fase de programación:

Se programan los componentes y se prueba la funcionalidad de la aplicación.

CIENCIA

TECNOLOGÍA

MATEMÁTICAS

GEOGRAFÍA/HISTORIA

IDIOMAS

LITERATURA

MÚSICA

OTROS

NIVEL DE ESTUDIOS

Esta actividad está preparada para...

12 - 14 AÑOS

14 - 16 AÑOS

OTROS

HERRAMIENTAS NECESARIAS

Materiales necesarios

- Cuaderno y lápices
- Ordenador
- Smartphone
- Instalación de la aplicación MIT AI2 Companion

ACTIVIDAD 3

DESARROLLO DE LA ACTIVIDAD

ÍNDICE

Paso 1. Creación de un boceto y planificación del proyecto
Crear un boceto y planificar el proyecto

Paso 2. Registrarse en App Inventor

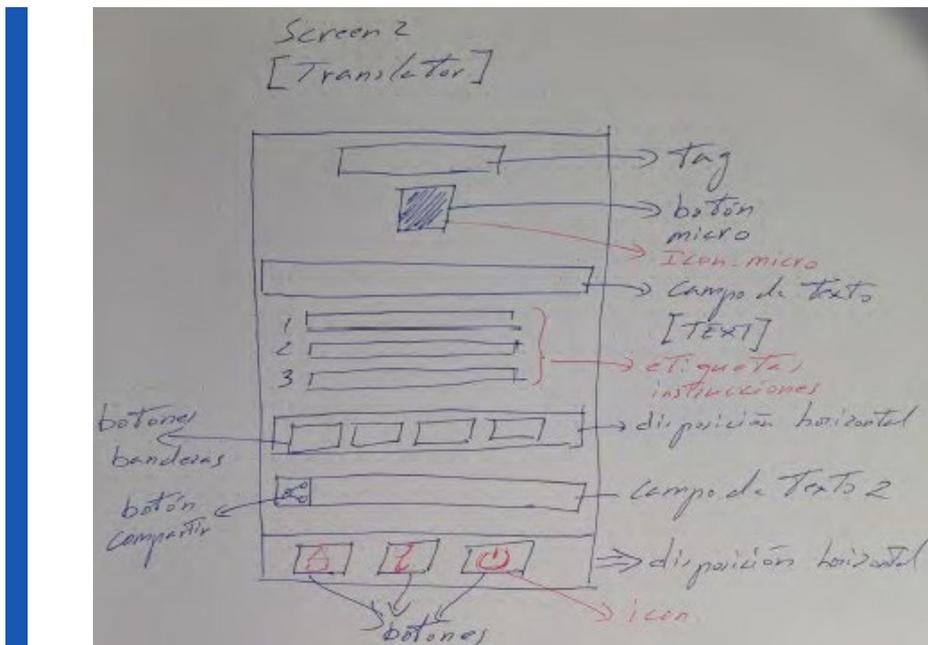
Paso 3. Diseño de las pantallas que componen la aplicación

Paso 4. Programación de los componentes de la aplicación

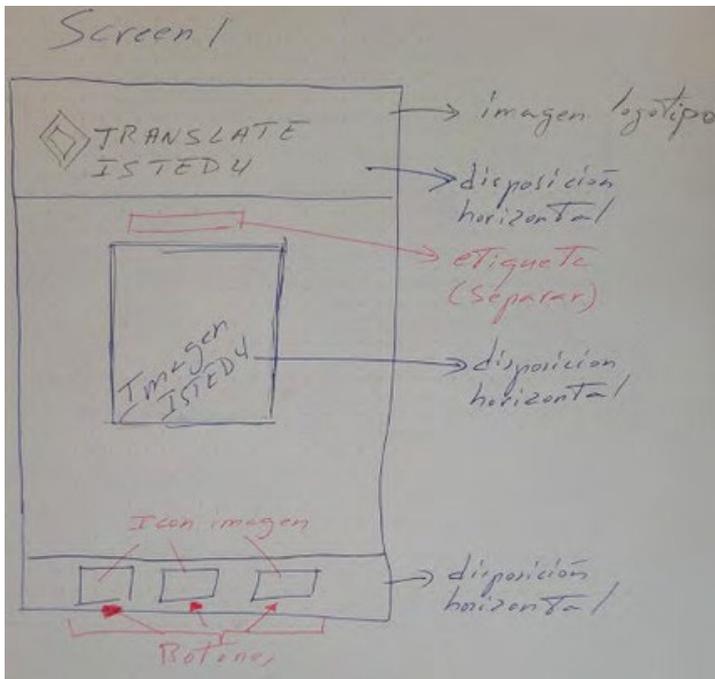
Paso 5. Probar los resultados e instalar la aplicación en un teléfono móvil

Paso 1. Creación de un boceto y planificación del proyecto

Para crear el prototipo de la aplicación es conveniente hacer un boceto dibujando el diseño y la colocación de los componentes.



ACTIVIDAD 3



En el diseño de estos bocetos es conveniente nombrar los componentes y la función que realizarán en el diseño de la aplicación.

Paso 2. Regístrate en App Inventor.

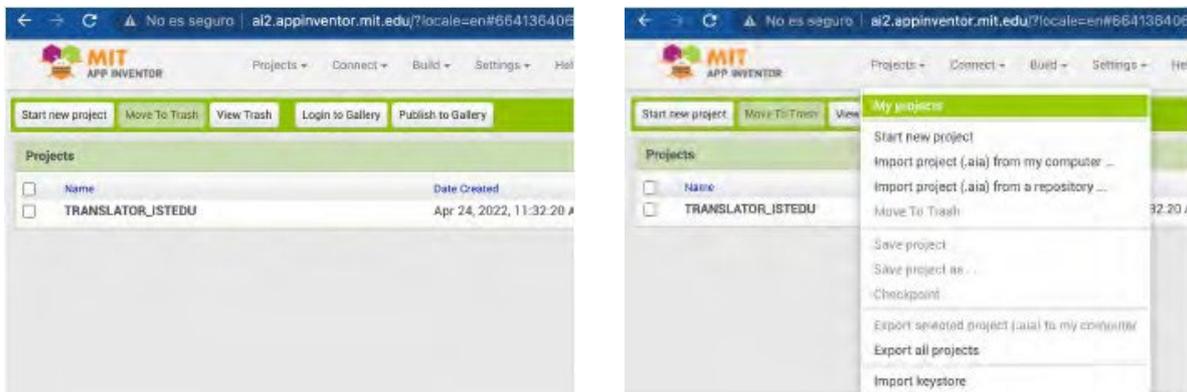
Para acceder a la plataforma, diríjase al siguiente enlace <https://appinventor.mit.edu/>

Accede a la web de la plataforma y haz clic en "Crear Apps".

Active Users today:	Active Users this week:	Active Users this month:	Registered Users:	Cour
111.5K	373.9K	1.1M	14.9M	1'

ACTIVIDAD 3

Se nos pedirá que iniciemos sesión con una cuenta de Google y desde ahí tendremos acceso directo a los proyectos en los que hemos trabajado. El último proyecto en el que hayamos trabajado se abrirá automáticamente. Para acceder a nuestros proyectos, debemos desplegar la pestaña "Proyectos" y acceder a nuestros proyectos.



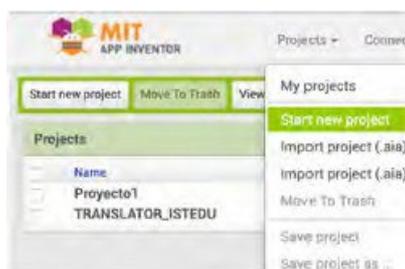
Paso 3. Diseño de las pantallas que componen la aplicación

La aplicación consta de tres pantallas:

1. Pantalla1. Esta es la pantalla de inicio y donde se encuentran los botones de acceso a las demás pantallas y a todos los se colocará la información que se considere pertinente para la solicitud.
2. Traductor. Esta es la pantalla donde se colocará el traductor.
3. Información. Utilizamos esta pantalla para colocar información sobre el proyecto

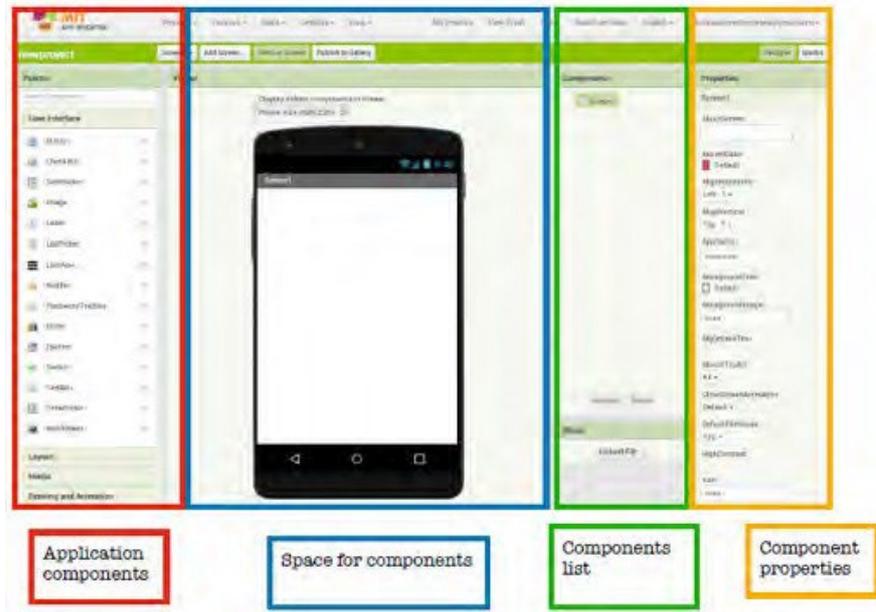
Pantalla "Pantalla1"

Para crear un nuevo proyecto, basta con seleccionar esta opción en la pestaña "Proyectos" seleccionando "Iniciar un nuevo proyecto".



ACTIVIDAD 3

Se nos pedirá que nombremos el proyecto y se abrirá una pantalla con estos componentes

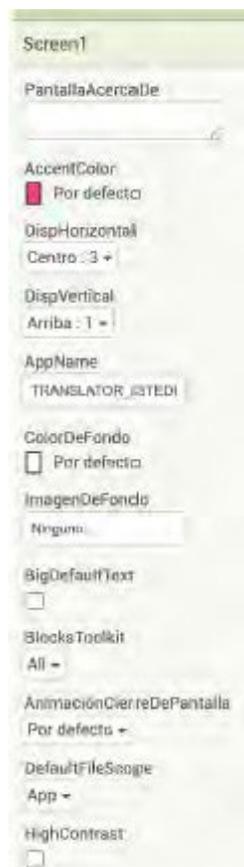


La pantalla tendrá las siguientes propiedades:

En este caso se ha decidido que la disposición horizontal colocará los componentes en el centro.

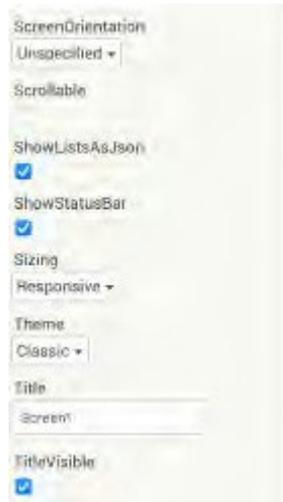
En la disposición vertical, los componentes se colocarán empezando por arriba.

Se ha decidido no poner ninguna imagen de fondo.



ACTIVIDAD 3

Desde esta casilla podemos cambiar el nombre de las pantallas.



En esta pantalla tendremos los siguientes componentes



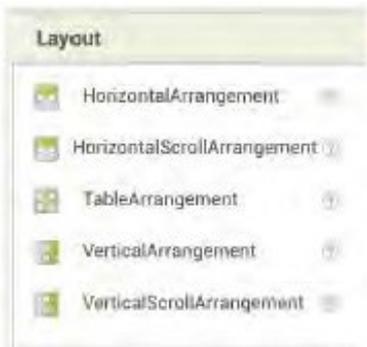
Horizontal arrangement with an image

Horizontal layout with picture

Horizontal arrangement with three buttons

ACTIVIDAD 3

Los diseños actúan como "cajas" en las que se colocan los componentes para que estén dispuestos de forma ordenada.

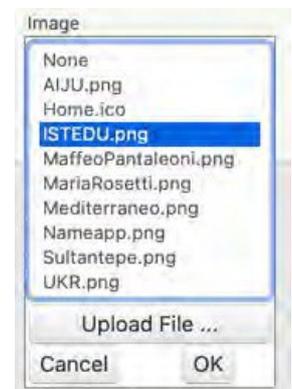


Para colocar una maqueta tenemos que abrir la pestaña "Maqueta" de la "Paleta" del y seleccione "Diseño horizontal". Arrástrelo al visor y colocarse en el centro y en la parte superior.

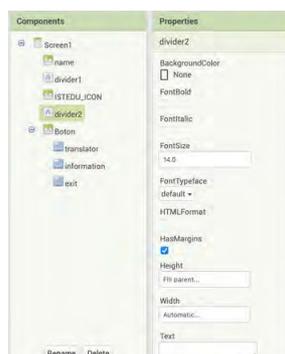
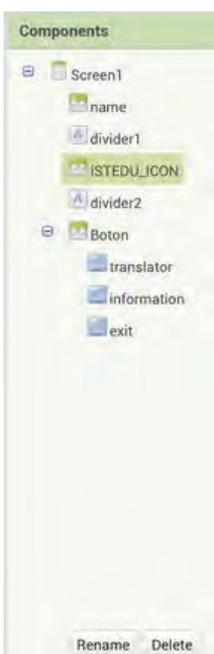
Las propiedades de esta disposición son las siguientes



Para colocar una imagen, basta con situar el cursor en el recuadro de la imagen y pulsar el botón izquierdo del ratón. A continuación, podemos cargar una imagen desde nuestro ordenador o utilizar una imagen ya cargada en el proyecto.



Es importante que renombramos los componentes que vamos a utilizar. Lo más útil es darles un nombre que esté asociado a la función que van a realizar. Esto es especialmente importante a la hora de colocar botones o componentes interactivos.



ACTIVIDAD 3

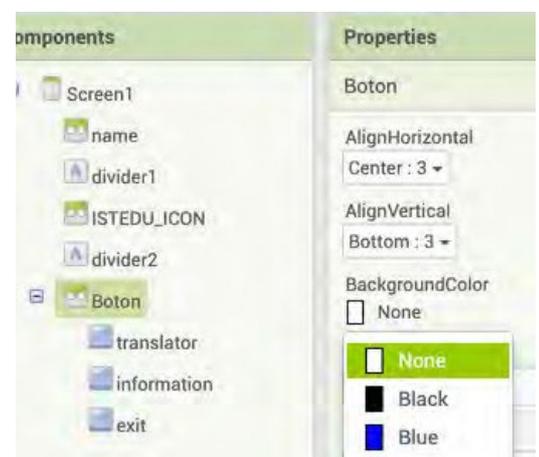
Para ajustar la posición en la disposición horizontal de la pantalla podemos utilizar el componente "Etiqueta" de la "Interfaz de usuario" y realizando ajustes en sus propiedades y eliminando el texto de la etiqueta podemos convertirla en un separador. En nuestro caso se han utilizado dos etiquetas como separadores.



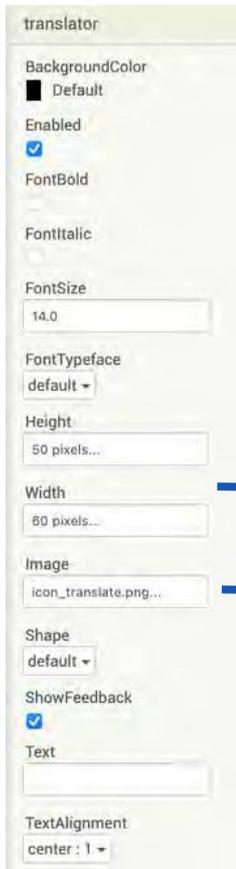
Para colocar los botones en esta pantalla en la parte inferior de la pantalla colocaremos un diseño horizontal y lo nombraremos "Botón" con estas propiedades.

Para colocar un botón dentro de un diseño horizontal, sitúe el diseño horizontal en el centro de la pantalla y elimine el color de fondo.

A continuación, desde "Interfaz de usuario", seleccionamos los botones y los arrastramos a la "disposición horizontal". Los botones se colocarán en el orden adecuado y podrán moverse dentro de la disposición horizontal.



ACTIVIDAD 3

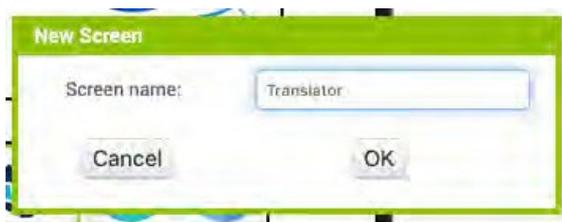


Las propiedades de los botones son las siguientes.

→ Se definen las dimensiones de los botones.

→ Seleccione una imagen para poner en el botón.

Antes de diseñar las nuevas pantallas tenemos que crearlas. Para crear una ventana, haga clic en el botón "Añadir pantalla" y dé un nombre a la nueva pantalla.

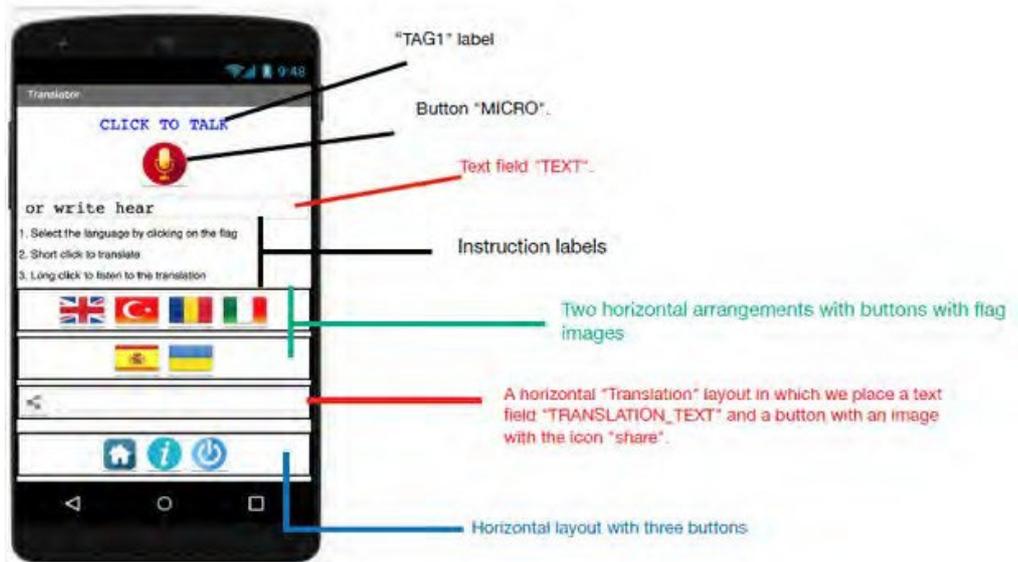


Las nuevas pantallas serán "Traductor" donde programaremos el traductor e "Información" donde colocaremos la información sobre el proyecto.

ACTIVIDAD 3

Pantalla "Traductor"

Los componentes de la pantalla son:



Las propiedades de los dos campos de texto



Estas dos propiedades son muy importantes

Debemos marcar esta opción "multilínea".

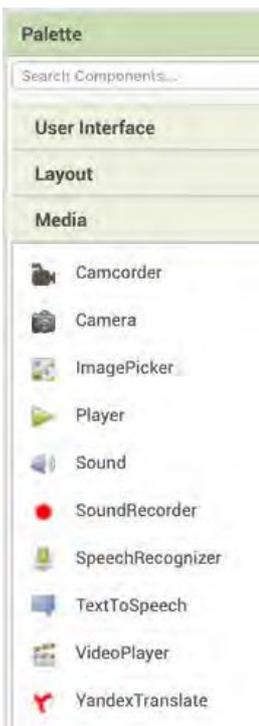
Este texto se ha colocado para indicar que, en lugar de hablar por el micrófono, puedes escribir en el espacio del campo de texto.

ACTIVIDAD 3

Las propiedades de las disposiciones horizontales



Componentes no visibles.

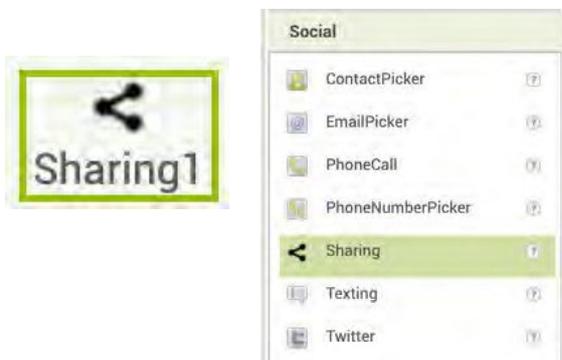


Hay una serie de componentes que se seleccionan en la pestaña "Medios" y que se utilizarán en la programación. Estos componentes se seleccionan y se arrastran a la pantalla que simula el teléfono. Una vez que los hemos arrastrado, se colocarán fuera de la pantalla porque son componentes que no se muestran. Los componentes que vamos a utilizar son:

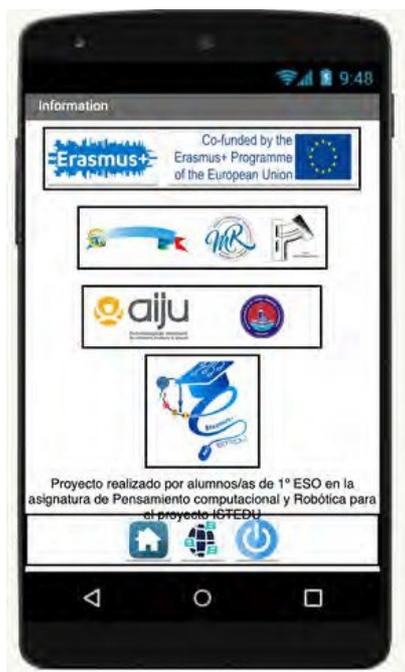


ACTIVIDAD 3

El componente Compartir 1 se encuentra en la pestaña "Social".



Pantalla "Información"



En esta pantalla hemos colocado simplemente diseños horizontales para los logotipos de los socios del proyecto, una etiqueta con información y un diseño horizontal con botones para volver a la pantalla de inicio, al traductor o para cerrar la aplicación.

Paso 4. Programación de los componentes de la aplicación Programación de los componentes de la aplicación

Para programar los distintos componentes de la aplicación, vaya a la sección "Bloques".



Los bloques de programación se encuentran en el lado izquierdo de la plataforma y están divididos en dos secciones.

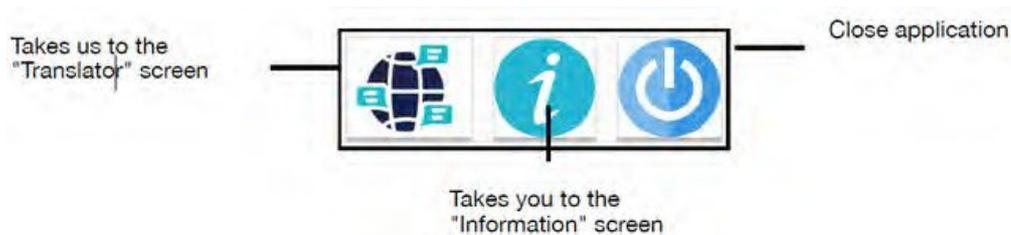
- Bloques genéricos (Integrados) que realizan funciones generales comunes a todos los componentes.

ACTIVIDAD 3



- Bloques específicos de cada uno de los componentes. Proporcionan funciones específicas de cada componente, botones, campos de texto, etiquetas....

En la primera pantalla "Screen1" sólo tenemos tres componentes interactivos correspondientes a los tres botones



Haga clic en la pestaña "Bloques" y acceda a la sección de programación, que estará vacía.



Seleccione el primer componente del panel izquierdo, que es el botón "traductor".

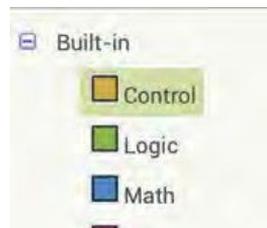
Desde el Visor, se muestran los bloques de programación.

ACTIVIDAD 3

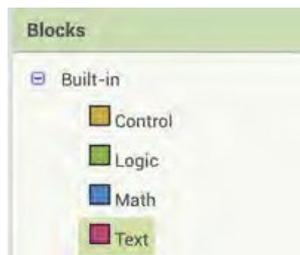


Seleccionamos este bloque y lo arrastramos al área de programación. Lo que estamos indicando es que pulsando sobre el botón se ejecutará un comando.

A continuación, desde los bloques comunes, desplegamos el bloque "control" y seleccionamos "abrir otra pantalla...".

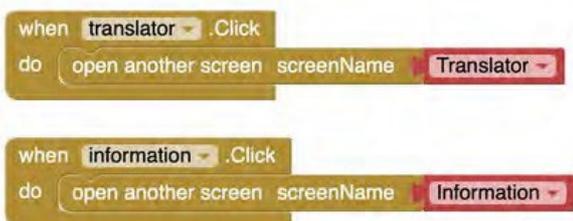


La orden que se ejecutará es que se abra otra pantalla. Sólo nos queda indicar qué pantalla queremos abrir. Para ello seleccionamos el bloque "Texto" en los bloques comunes.



Seleccionamos el bloque  donde escribimos el nombre de la pantalla que queremos abrir.

Es muy importante que introduzca el nombre exacto de la pantalla que desea abrir. Por último, sólo queda encajar los bloques de programación



Este procedimiento se se realizado en el botón que abre la pantalla "Información".



Para programar el botón "Cerrar aplicación", seleccione el bloque "Cerrar aplicación" en "Control".

ACTIVIDAD 3

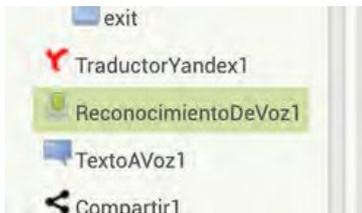
Pantalla "Traducción"

El traductor se programará desde la página "Traducción".



La programación comienza utilizando el botón "Micro". Desde la sección Bloques seleccionamos el botón micro y utilizamos el bloque de activación con un clic.

El siguiente paso es seleccionar el componente no visible "SpeechRecognition1" y activar el reconocimiento de voz con este bloque de programación.



Este sería el resultado de enlazar los bloques. Al pulsar el botón "Micro", se activa el reconocimiento de voz, que nos dirá qué hacer con el texto reconocido.



Del componente de reconocimiento de voz, seleccionamos este bloque

A continuación, desplegamos las funciones del primer bloque de texto "TEXTO" y elegimos este bloque.



Lo que estamos indicando es que una vez que el reconocimiento de voz ha reconocido un texto lo coloque en el Campo de Texto

ACTIVIDAD 3



Desde el bloque Reconocimiento de voz, desplegamos la opción "Resultado" y seleccionamos



El resultado final será:



Ya hemos activado el micrófono y colocado el resultado de reconocer lo que se ha hablado en un campo de texto. Ahora activamos el traductor y le decimos que coloque el resultado de la traducción en el campo de texto 2

"TRANSLATION_TEXT".

Para activar el traductor seleccionamos el traductor Yandex y del campo de texto 2 seleccionamos el campo de texto "TRANSLATION_TEXT". El resultado será



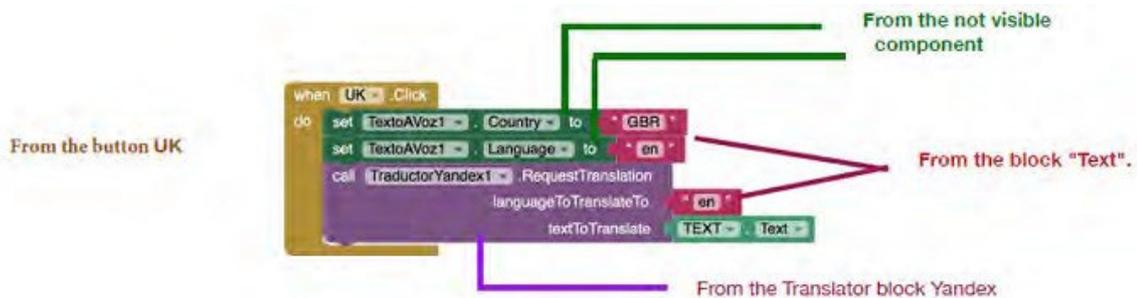
Lo que vamos a hacer ahora es indicar a qué idioma queremos que se haga la traducción. Para ello utilizaremos los botones donde hemos colocado las banderas de los países.

La traducción se activará de dos formas distintas.

- Con un clic corto, el resultado de la traducción se escribirá en el campo Texto 2.
- Con un clic largo podrás escuchar el resultado de la traducción en el idioma seleccionado. Es importante tener en cuenta que esta posibilidad dependerá de la configuración de idioma de cada teléfono móvil.

Para traducir un texto al inglés, sigue estos pasos:

ACTIVIDAD 3



Para las abreviaturas de lenguas y países se utilizan los códigos ISO.

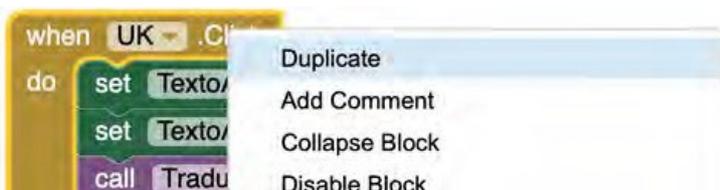
[Enlace para los idiomas ISO](#) 

[Enlace al código de país ISO](#) 

Para activar que la traducción se escuche en el idioma seleccionado colocamos estos bloques de programación

```
when UK .LongClick
do
  call TextoAVoz1 .Speak
  message TRANSLATION_TEXT .Text
  set TextoAVoz1 .Country to "GBR"
  set TextoAVoz1 .Language to "en"
```

Una vez que hemos programado el botón del Reino Unido, podemos copiar los códigos y simplemente cambiar el nombre del botón y los códigos ISO de los países e idiomas. Haremos esta operación para tantos idiomas como botones con banderas hayamos colocado.



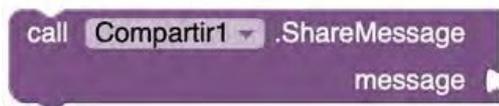
Sólo queda activar el botón "Compartir". Este botón permite copiar el texto traducido y pegarlo en nuevas redes sociales, un procesador de textos o un correo electrónico.

ACTIVIDAD 3



Desde el botón "Compartir" seleccionamos la opción Hacer clic.

En la no visible componente "Share1" activamos la función compartir



Por último, le indicamos qué texto queremos

compartir.

Los botones de la parte inferior se programan del mismo modo que en la pantalla "Pantalla1".

Pantalla "Información"

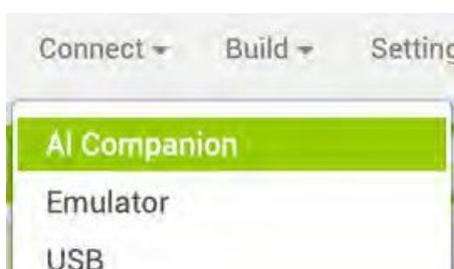
Esta pantalla no requiere más programación que los botones que permiten cambiar de pantalla o cerrar el aplicación.

Paso 5. Compruebe los resultados e instale la aplicación en un teléfono móvil

Para poder comprobar cada paso en el desarrollo de la aplicación, es necesario tener la aplicación instalada en el teléfono móvil.



En la pestaña "Conectar", selecciona la opción AI Companion. Es importante que el ordenador y el teléfono móvil estén conectados a la misma red WIFI.



ACTIVIDAD 3

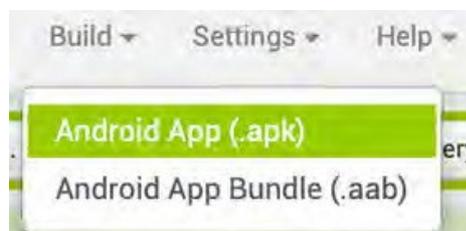
Cuando hagas tu selección, se abrirá una página con un código QR.



Ahora desde el móvil abriremos la aplicación MIT AI2 Companion y seleccionaremos "escanear código QR" y escanearemos el código QR.

Se abrirá una ventana indicando el nivel de progreso. Una vez finalizado, la aplicación estará operativa en el teléfono móvil. Hay que tener en cuenta que no se ha instalado ninguna aplicación en el teléfono, sólo estamos realizando una emulación de la App.

El botón de opción "Cerrar la aplicación" no funciona en modo emulación. Para instalar la aplicación en un smartphone, en la pestaña Generar, seleccione Aplicación Android (.apk).



Aparecerá una ventana indicando el progreso de la compilación de la App.



ACTIVIDAD 3

Cuando termine, se generará un archivo apk, que podrá descargarse y enviarse al smartphone.



RECURSOS



EVALUACIÓN DEL ALUMNO

A evaluar:

- Diseñar una estrategia global del proyecto
 - Para especificar bien la función de la App
 - Diseño de un boceto con las pantallas y componentes de la App
 - Utilización de los elementos básicos de la interfaz de usuario
 - Utilización correcta de los elementos del
- Layout Descomposición del proyecto en tareas más sencillas
- Diseño correcto y funcional de las pantallas de la App
- Programación de los componentes de la App
- Revisión de las versiones del proyecto y corrección de posibles errores
- Funcionalidad del proyecto
- Posibilidad de ampliar el sitio aplicación con la incorporación de nuevas funcionalidades

BIBLIOGRAFÍA

- MIT App Inventor: <http://appinventor.mit.edu/>
- Guía de iniciación a App Inventor: <http://codeweek.eu/resources/spain/guia-iniciacion-appinventor.pdf>
- Tutorial MIT App Inventor: <https://rominirani.com/tutorial-mit-app-inventorfirebase-4be95051c325>
- App Inventor 2 (en español): <http://kio4.com/appinventor/>
- Curso App Inventor: <https://www.youtube.com/watch?v=sQ2EmGNp2U4>

ACTIVIDAD 3

ADAPTACIÓN A OTROS NIVELES

El desarrollo de aplicaciones con App Inventor permite crear aplicaciones que conectan con tarjetas Arduino y actúan como controladores vía wifi o bluetooth para robots o domótica entornos.

También es posible crear aplicaciones que incorporen IA

MÁS INFORMACIÓN

[Enlace a la carpeta de Google Drive en la que se ha cargado la aplicación](#)



[Enlace a la tabla de idiomas ISO](#)



[Enlace para obtener el código ISO de los países](#)



ACTIVIDAD 4

TÍTULO Partido de fútbol

RESUMEN

STEM: Ciencia, Tecnología, Ingeniería y Matemáticas. Es mucho material complejo para un solo acrónimo. Afortunadamente para los más jóvenes, la robótica hace que STEM sea fácil y divertido de aprender.

La pasión por STEM y POR LOS ROBOTS se puede materializar a través de juegos que, de forma amena y atractiva, aumentan la motivación de los alumnos por aprender y practicar y por poner en práctica numerosas actividades y proyectos.

Con tantas opciones para lo que los niños pueden aprender con la robótica, es fácil ver los beneficios basados en habilidades de utilizarla como hobby.

Además, hay más buenas noticias: Aprender los entresijos de la robótica significa desarrollar aptitudes profesionales en el mundo real.

¿Por qué no dar el paso hoy mismo y seguir estos pasos?

Los principales beneficios educativos de la robótica para niños y adolescentes se deben a la naturaleza interactiva y táctil de este campo, además de la disponibilidad de clubes escolares, sencillos juguetes robóticos para casa y cursos de robótica en línea.

La robótica ofrece excelentes oportunidades de trabajo en equipo y colaboración para diseñar y construir un robot que se enfrente a sus oponentes en una serie de desafíos.

Por eso hicimos un pequeño partido de fútbol entre dos equipos utilizando 1 robot por cada grupo de alumnos

AUTOR/S

SCOALA GIMNAZIALA MARAI ROSETTI

FECHA 30/11/2022

VERSIÓN 1

ACTIVIDAD 4

OBJETIVOS DIDÁCTICOS

Los alumnos aprenden:

- Controlar un robot
- Conceptos de programación
- Transición progresiva desde cero a Arduino por comparación siendo una forma muy útil de aprender
- Qué es mbot, cómo funciona
- Reconoce el programa Mblock
- Cómo utilizar keyes BT car, Tank car apk y keyes 4wd.apk

CIENCIA

TECNOLOGÍA

MATEMÁTICAS

GEOGRAFÍA/HISTORIA

IDIOMAS

LITERATURA

MÚSICA

OTROS

NIVEL DE ESTUDIOS

Esta actividad está preparada para...

12 - 14 AÑOS

14 - 16 AÑOS

OTROS

HERRAMIENTAS NECESARIAS

Mbot(Bluetooth)

Mblock Programa

Mbot Remote

Poliestireno para delimitar el terreno de juego

2 Robots

2 teléfonos

Software para ANDROID e IPHONE

DESARROLLO DE LA ACTIVIDAD

A continuación se describe paso a paso la configuración del proyecto

1. Las piezas de poliestireno se midieron y cortaron para fijarlas al suelo. Representan el límite de movimiento del robot por el suelo. Es preferible fijar bien las piezas de poliestireno, de lo contrario los robots las moverán durante el juego.
2. Las puertas estaban hechas de papel A4
3. La pelota utilizada fue una de tenis, pero se puede utilizar cualquier pelota pequeña pero con algo de peso para poder manejarla con más facilidad.

En el proyecto controlamos 3 robots diferentes.

- La primera y más sencilla es la aplicación Mbot controlapplication, que descargamos de la Play Store. Tras establecer una conexión bluetooth entre el mbot y el dispositivo móvil, podrás manejar fácilmente el robot.
- La segunda es codificar Mbot con el programa Mblock. Puede funcionar con mblok 3.0 y versiones superiores. Hay bloques de código en Mblock que permiten al robot girar directamente a la derecha-izquierda. Estos han sido utilizados. Preferimos esto porque trabajamos en un área amplia; pero aquellos que trabajan en un área más estrecha también pueden proporcionar rotación diferenciando las velocidades del motor derecho e izquierdo. Por ejemplo, cuando la velocidad del motor M1 es 100 y la velocidad del motor M2 es 255, Mbot girará en una dirección. En esta etapa, utilizamos las teclas del ordenador (wasd y espacio)
- La tercera es un poco más difícil pero más instructiva para nuestros alumnos. Escribimos nuestra propia aplicación, montamos el robot nosotros mismos, establecemos la conexión entre el robot y la aplicación móvil nosotros mismos.

ACTIVIDAD 4

RECURSOS

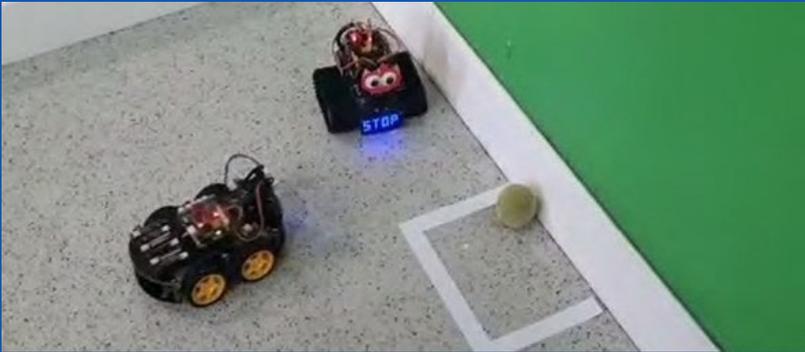


ACTIVIDAD 4



```
when green flag clicked
  forever loop
    if key w pressed? then
      run forward at speed 50
    if key s pressed? then
      run backward at speed 50
    if key d pressed? then
      turn right at speed 50
    if key a pressed? then
      turn left at speed 50
    if key space pressed? then
      run forward at speed 0
```

ACTIVIDAD 4



EVALUACIÓN DEL ALUMNO

Áreas en las que nuestros alumnos se evaluarán a sí mismos:

1. ¿Puedo utilizar el software Mblock para Mbot?
2. ¿Puedo modelar con Tinkercad?
3. ¿Puedo utilizar correctamente los materiales del proyecto y crear el terreno adecuado?

BIBLIOGRAFÍA

<https://ide.mblock.cc/>

<https://appinventor.mit.edu/>

ACTIVIDAD 4

ADAPTACIÓN A OTROS NIVELES

Este proyecto puede considerarse de nivel principiante-intermedio para alumnos de quinto y sexto curso.

Cuando Mbot ve un obstáculo se detiene por el sensor de distancia y luego es dirigido por el teléfono para poder golpear la pelota

ACTIVIDAD 5

TÍTULO Señales de tráfico con Maqueen

RESUMEN

El uso de la tecnología se está convirtiendo en un aspecto cada vez más importante en

vida cotidiana. Se diseñan nuevas figuras profesionales y nuevas herramientas de trabajo para llevar a cabo tareas innovadoras.

La tarea de la escuela es actualizarse a esta realidad, pero sobre todo proporcionar a los alumnos las primeras bases y prepararlos para el futuro. Hoy veremos, pues, la importancia de la codificación en la escuela.

Estar a la altura de los tiempos es la clave para poder captar el interés de los alumnos y hacerles participar en el proceso de aprendizaje.

El cambio y la innovación son, por tanto, una responsabilidad real también para las escuelas. El primer objetivo de la codificación es enseñar a los niños a desarrollar una mente elástica y proyectada a soluciones eficaces de problemas sencillos

Los alumnos construyen un camino y programan el robot que sigue el camino respetando las señales de tráfico que encuentra.

AUTOR/S

FECHA 14/09/2022

VERSIÓN 1

IPS Maffeo Pantaleoni

OBJETIVOS DIDÁCTICOS

- Saber resolver problemas
- Saber tomar decisiones
- Creatividad
- Sentido crítico
- Autoconciencia
- Habilidades
- interpersonales
- Comunicación eficaz
- Gestión de las emociones

ACTIVIDAD 5

- Gestión del estrés
- Empatía
- Experiencia de trabajo en equipo
- Fomentar la integración de estudiantes discapacitados;
- Fomentar la integración de estudiantes extranjeros;
- Fomentar el espíritu de colaboración;
- Estimular el pensamiento creativo;
- Aumentar la capacidad de toma de decisiones, el sentido de la responsabilidad y la autoestima;
- Desarrollar la capacidad de analizar y resolver problemas;
- Mejorar un lenguaje de programación.

CIENCIA

TECNOLOGÍA

MATEMÁTICAS

GEOGRAFÍA/HISTORIA

IDIOMAS

LITERATURA

MÚSICA

**OTROS: DERECHO
Y EDUCACIÓN CÍVICA**

NIVEL DE ESTUDIOS

Esta actividad está preparada para...

12 - 14 AÑOS

14 - 16 AÑOS

OTROS: 16 - 18 AÑOS

HERRAMIENTAS NECESARIAS

- Robot Maqueen
- Cartón
- Ordenador
- Programas de diseño 3D
- Impresora 3D

ACTIVIDAD 5

```
Forever
  if Giro = 0 then
    if Button A is pressed then
      radio send number 1
      show number 1
    if Button B is pressed then
      radio send number 2
      show number 2
    if Logo is pressed then
      radio send number 3
      show number 3
    if acceleration (mg) < -500 then
      radio send number 4
      show leds
    if acceleration (mg) > 500 then
      radio send number 5
      show leds
    if acceleration (mg) < -500 then
      radio send number 6
      show leds
    if acceleration (mg) > 500 or -500 < acceleration (mg) < -500 or acceleration (mg) < 500 then
      show leds

on start
  radio set group 33
  set Giro to 0

on radio received receivedNumber
  if receivedNumber = 1 then
    set Giro to 0
```

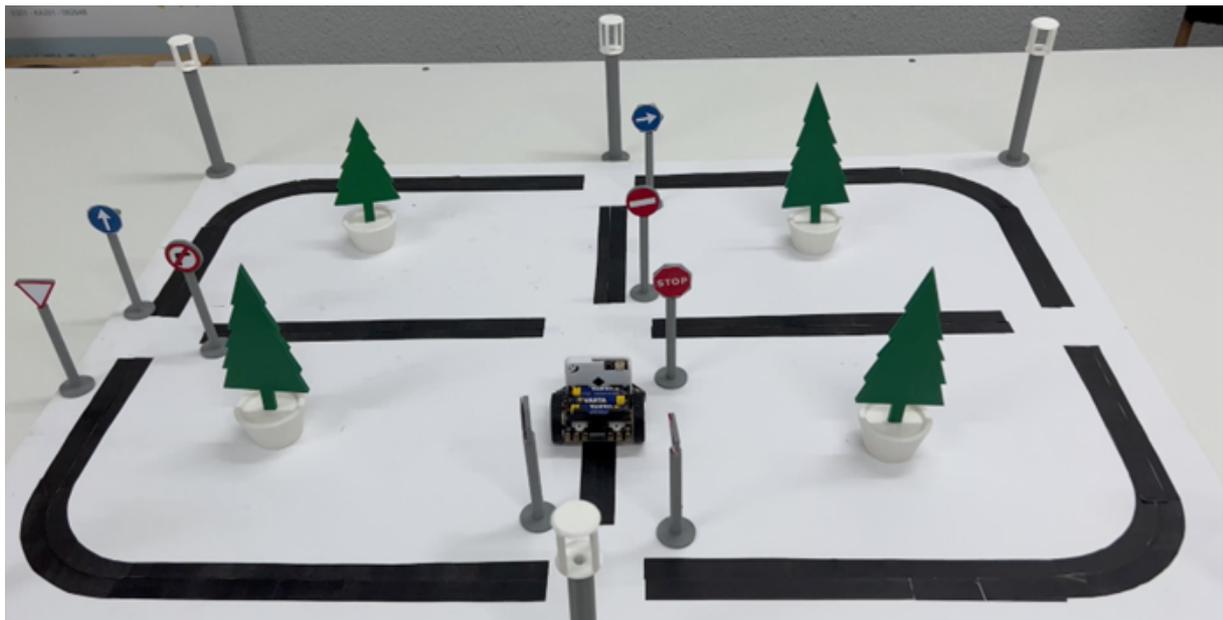
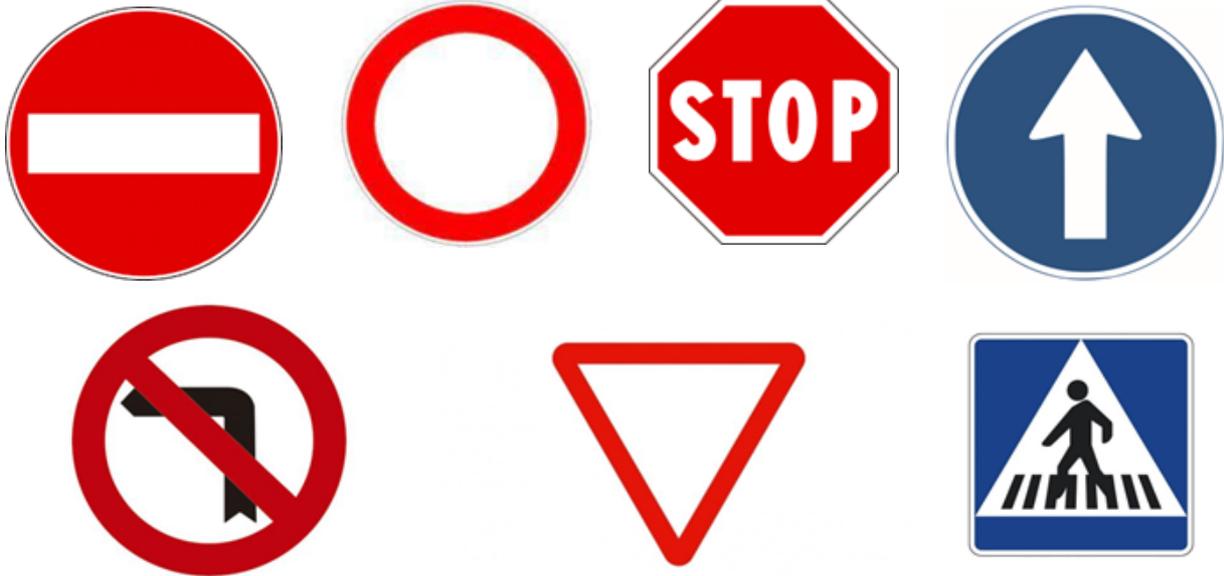
ACTIVIDAD 5

- Programar emisor

```
on robot receiveDistanceFromSensor
  if receiveNumber == 1 then
    Contrast == 2 then
      while read left* line tracking sensor == 1 and read right* line tracking sensor == 1
        motor left* move forward* at speed*
        motor right* move forward* at speed*
        pause 100
      set Control* to 2
    if receiveNumber == 1 then
      Contrast == 2 then
        while read left* line tracking sensor == 1 and read right* line tracking sensor == 1
          motor right* move forward* at speed*
          motor left* move forward* at speed*
          pause 100
        set Control* to 2
      if receiveNumber == 1 then
        Contrast == 2 then
          while read left* line tracking sensor == 1 and read right* line tracking sensor == 1
            motor right* move forward* at speed*
            motor left* move forward* at speed*
            pause 100
          set Control* to 2
        if receiveNumber == 1 then
          Contrast == 2 then
            LEDlight right* turn ON*
            pause 100
            LEDlight right* turn OFF*
            pause 100
          if receiveNumber == 1 then
            Contrast == 2 then
              LEDlight left* turn ON*
              pause 100
              LEDlight left* turn OFF*
              pause 100
            pause 100
          if not read left* line tracking sensor == 1 and read right* line tracking sensor == 1 then
            set Control* to 2
  on robot receiveDistanceFromString
  if receiveDistance == 100 then
    LEDlight left* turn ON*
    pause 100
    LEDlight left* turn OFF*
    pause 100
  on start
  public set pin*
  set Contrast* to 2
  forward
  if Contrast == 2 then
    if read left* line tracking sensor == 1 and read right* line tracking sensor == 1 then
      motor all* move forward* at speed*
    if read left* line tracking sensor == 1 and read right* line tracking sensor == 1 then
      motor right* move forward* at speed*
      motor left* move forward* at speed*
    if read left* line tracking sensor == 1 and read right* line tracking sensor == 1 then
      motor left* move forward* at speed*
      motor right* move forward* at speed*
    if read left* line tracking sensor == 1 and read right* line tracking sensor == 1 then
      motor all* stop
      set Control* to 2
      public send number*
  on robot receiveDistanceFromString
  if receiveDistance == 100 then
    LEDlight left* turn ON*
    pause 100
    LEDlight left* turn OFF*
    pause 100
```

ACTIVIDAD 5

Señales de tráfico

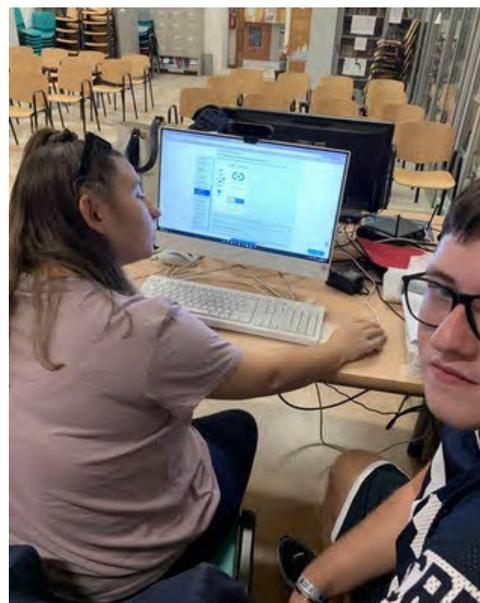


ACTIVIDAD 5

RECURSOS



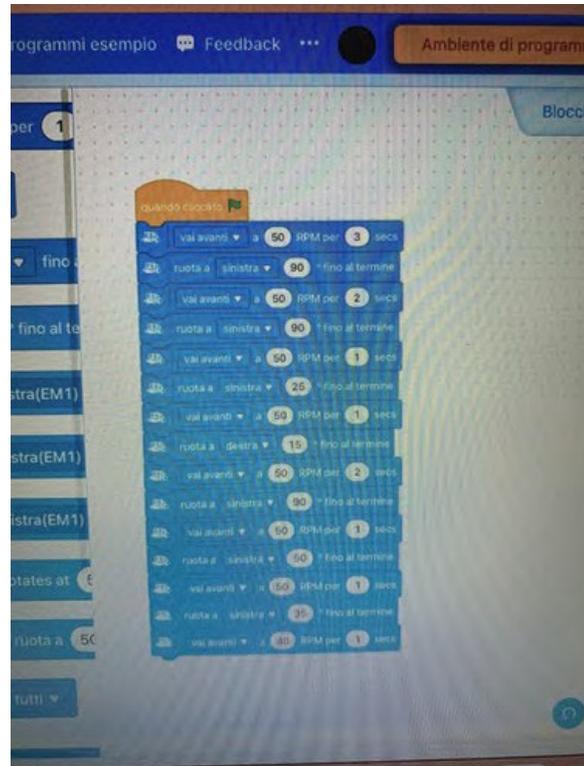
Señales de tráfico: prohibido el paso, prohibido el tránsito, stop, prohibido aparcar



ACTIVIDAD 5



ACTIVIDAD 5



EVALUACIÓN DEL ALUMNO

Test sobre el significado de las señales de tráfico y algunos supuestos prácticos.

Programación del Mcqueen para que realice todos los movimientos y de la placa que controla los movimientos.

BIBLIOGRAFÍA

Señales de tráfico
Maqueen siguelineas
Mlicrobit radio

ACTIVIDAD 5

ADAPTACIÓN A OTROS NIVELES

El robot también puede utilizarse con alumnos de primer ciclo de secundaria (11-13 años).

ACTIVIDAD 6

TÍTULO Diseño de una versión del videojuego Space Invaders utilizando Mblock

RESUMEN

El juego consiste en mover un sprite (nave) en el plano horizontal y eliminar una serie de objetos (invasores) que se mueven en formación de izquierda a derecha y hacia abajo. El sprite de la nave debe evitar los disparos de un sprite que se mueve en el plano horizontal superior. Este sprite puede ser eliminado si recibe tres impactos. También debe evitar entrar en contacto con un tercer sprite que tiene un movimiento aleatorio y que eventualmente intentará golpear al sprite de la nave.

El juego dispone de defensas contra objetos que permiten ocultar el sprite de la nave. Estos sprites se "destruirán" cuando sean alcanzados por un disparo o entren en contacto con objetos invasores.

El juego termina cuando todos los sprites invasores son eliminados o cuando la nave controlada por el jugador es eliminada. Este sprite de nave es eliminado cuando recibe tres impactos o cuando los sprites invasores alcanzan una determinada posición en el eje Y.

AUTOR/S

FECHA 26/03/2022
VERSIÓN 1

IES Mediterráneo

OBJETIVOS DIDÁCTICOS

Profundizar en el desarrollo de la programación con pseudocódigo.

Trabajaremos aspectos relacionados con:

- Cargar sprites
- Movimientos
- Uso de variables
- Uso de clones
- Utilización de sonidos
- Creación de diálogo

ACTIVIDAD 6

Estrategia de programación: Se utilizará una estrategia basada en el Pensamiento Computacional. Dividiremos el problema global, la creación de un videojuego tipo "Space Invaders", en problemas más pequeños y se dará solución a cada uno de estos problemas para obtener finalmente una solución al problema general.

El proyecto puede ampliarse instalando dispositivos controlados por una placa Arduino.

Problema general: Creación de un videojuego tipo "Space Invaders" con una estética basada en la saga Star Wars.

Problemas parciales:

1º Creación de fondos 2º

Acceso al juego

4º Creación de la nave principal (Ala X)

5º Creación de la nave atacante (Estrella de la Muerte)

6º Creación de nave secundaria (Starfighter TIE Fighter

) 7º Interacción de sprites

8º Incorporar sonido

9º Fin del juego

CIENCIA

TECNOLOGÍA

MATEMÁTICAS

GEOGRAFÍA/HISTORIA

IDIOMAS

LITERATURA

MÚSICA

OTROS

NIVEL DE ESTUDIOS

Esta actividad está preparada para...

12 - 14 AÑOS

14 - 16 AÑOS

OTROS

ACTIVIDAD 6

HERRAMIENTAS NECESARIAS

Equipamiento necesario

- Ordenador
- Mblock versión 3.11

Material necesario para ampliar el

- proyecto Placa Arduino
- LEDs de la protoboard
- Cables de conexión
- Resistencias
- Yoystick

DESARROLLO DE LA ACTIVIDAD

ÍNDICE

Paso 1. Creación de fondos Creación de fondos

Paso 2. Creación de la nave principal (Ala X) y acceso al juego

Paso 3. Creación de clones (Soldados Imperiales)

Etapas 4. Creación de defensas

Paso 5. Creación de la nave atacante (Estrella de la Muerte)

Paso 6. Creación de nave secundaria (Starfighter TIE Fighter)

Paso 7. Interacción de sprites

Paso 8. Incorporar el sonido

Paso 1: Creación de fondos

Antes de empezar a programar el videojuego tenemos que limpiarlo de objetos.

El juego requiere cuatro fondos:

1. Antecedentes de la presentación
2. Introducción
3. Experiencia en desarrollo de juegos
4. Antecedentes de perder el juego (Game Over)
5. Ganar el fondo del juego ("Victoria")

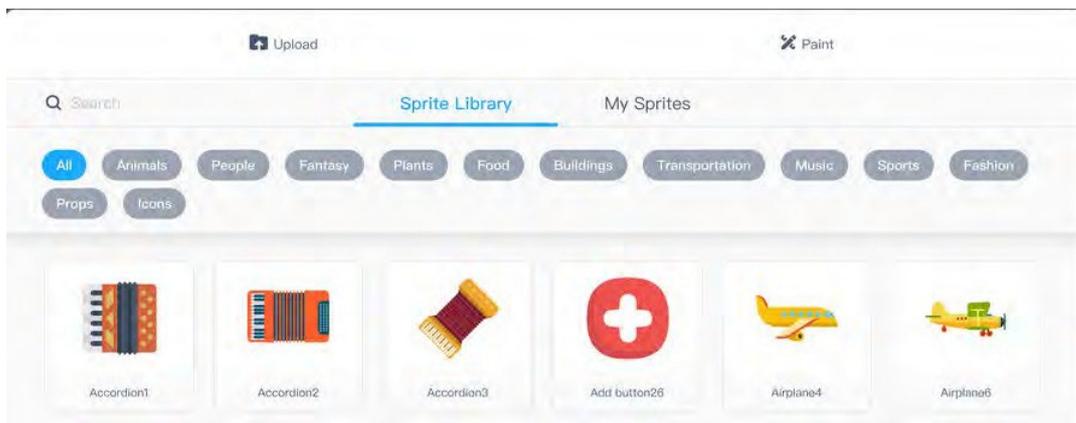
ACTIVIDAD 6

Empezaremos subiendo el fondo de presentación del juego. El resto de fondos se realizan siguiendo el mismo procedimiento.

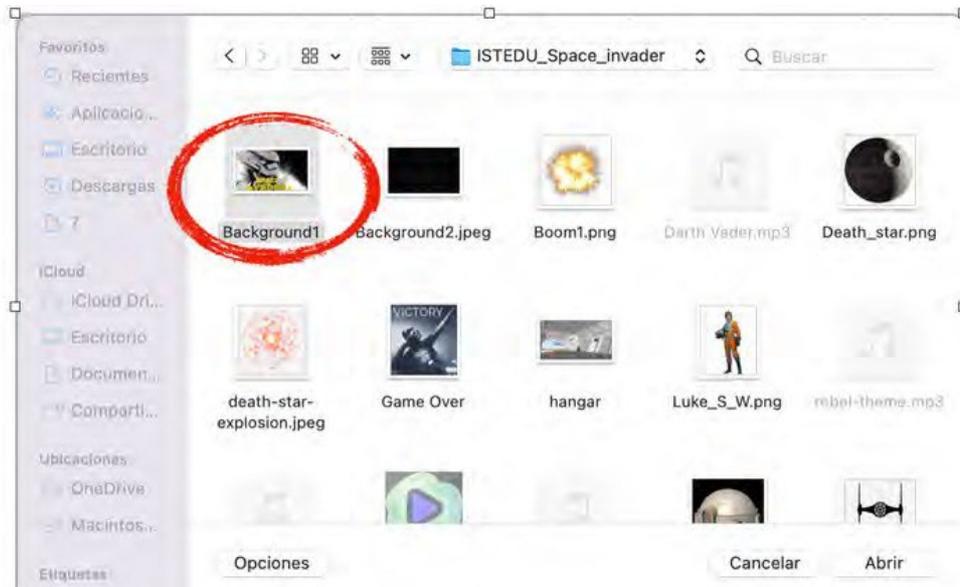
Para subir el fondo de la presentación necesitaremos una imagen que tengamos guardada en nuestro ordenador.

(En el material complementario se añadirá una dirección url a una carpeta compartida donde se incorporan las imágenes y objetos del proyecto).

En la pestaña "Fondo", haz clic en "Trajes" y luego en "Añadir un fondo".



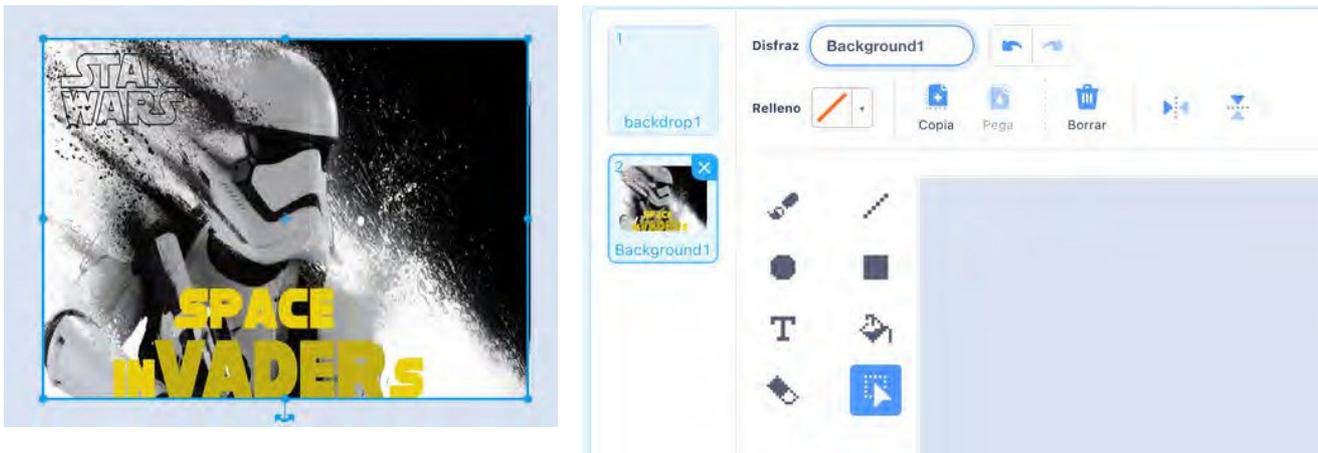
En la pantalla que se abre, haz clic en "Cargar" y selecciona la imagen de tu ordenador.



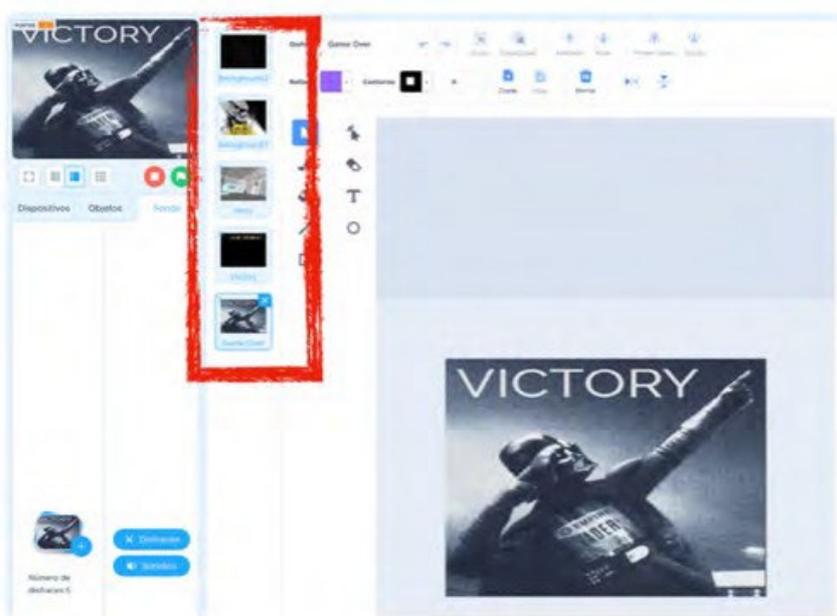
Seleccione la imagen y haga clic en "Abrir" y "Aceptar".

ACTIVIDAD 6

Para ocupar todo el espacio del fondo, debemos seleccionar la imagen y seleccionar toda la imagen para ampliarla y que ocupe toda la superficie. Finalmente le daremos el nombre de "Fondo1" y eliminaremos el fondo que aparece por defecto "fondo1".



El resto de los fondos se hacen de la misma manera que los anteriores. Es muy importante guardar los fondos con nombres que faciliten la programación. Como vemos, tenemos 5 fondos que utilizaremos en la programación del videojuego.



ACTIVIDAD 6

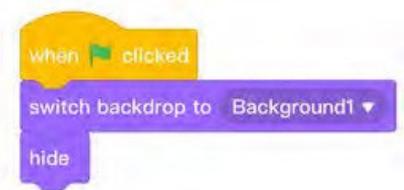
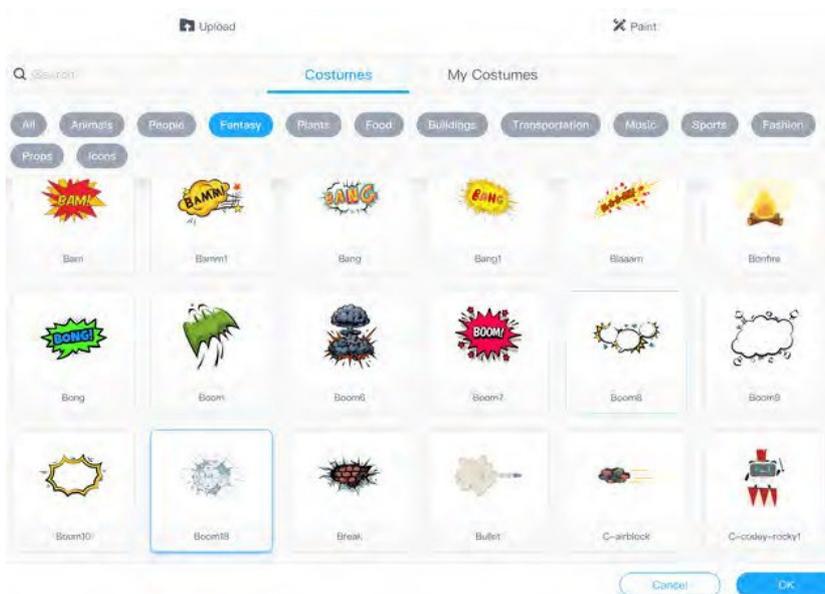
Paso 2. Creación de la nave principal (X-Wing) y acceso al juego.



Para empezar a programar el juego tendremos que subir el Background1 y el sprite del X-Wing. Para subir el sprite del X-Wing, que será la nave que controlaremos, sólo tenemos que ir a la sección "Sprites" y hacer clic en "Añadir



Subiremos el sprite y además le añadiremos un disfraz seleccionando una explosión del repertorio de sprites que ofrece M-Block. Para ello haremos clic en "Añadir disfraz" y del repertorio seleccionaremos el que mejor se adapte a nuestras necesidades.



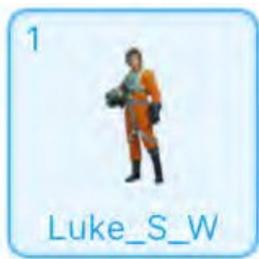
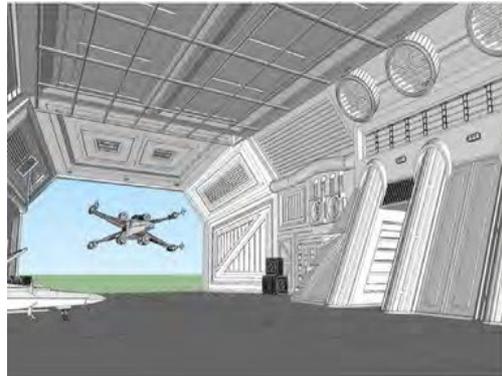
Desde el área de programación en el sprite del X-Wing iniciaremos el juego con este sencillo programa. Lo que indicamos es que al pulsar "Bandera verde" se inicie el juego con el fondo "Bacground1" y se oculte el sprite

ACTIVIDAD 6

```
when clicked
wait 4.5 seconds
switch backdrop to Intro
```

A continuación cambiaremos a la pestaña "Fondo" y colocaremos este programa para que después de 4,5 segundos cambie al fondo "Intro".

Introducción



El siguiente paso es subir el sprite "Luke_S_W".

El primer programa a cargar con cada sprite es indicar cuando queremos que esté oculto o visible. En este caso indicaremos que al pulsar "bandera verde" y en el fondo "Fondo1" estará oculto.

```
when clicked
hide

when backdrop switches to Background1
hide
```

A continuación indicaremos que cuando el fondo cambie a "Enter" el sprite

- La voluntad es visible
- Fijaremos un tamaño
- Le daremos unas coordenadas en los ejes X e Y para situarlo en el lugar que queramos del fondo.

```
cuando el fondo cambie a Intro
muestra
fija tamaño a 90 %
ve a x: -121 , y: -29
```

ACTIVIDAD 6

El siguiente paso es entablar un diálogo con el jugador para iniciar el juego. Desde la sección "Apariencias" incluiremos una frase introductoria



De la sección "Sensing" incluiremos una pregunta



Para continuar con una nueva pregunta en la que podamos incluir la respuesta del jugador necesitaremos unir cuatro bloques que se encuentran en la sección "Sensores" y "Operadores".

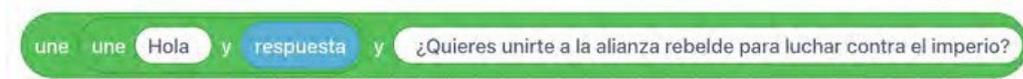


El objetivo es tomar la respuesta con el nombre del jugador y hacerle una pregunta directamente. Para ello enlazaremos los bloques de la siguiente manera.

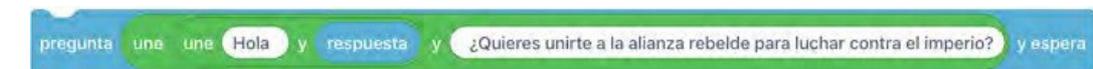
1º



2º



3ª



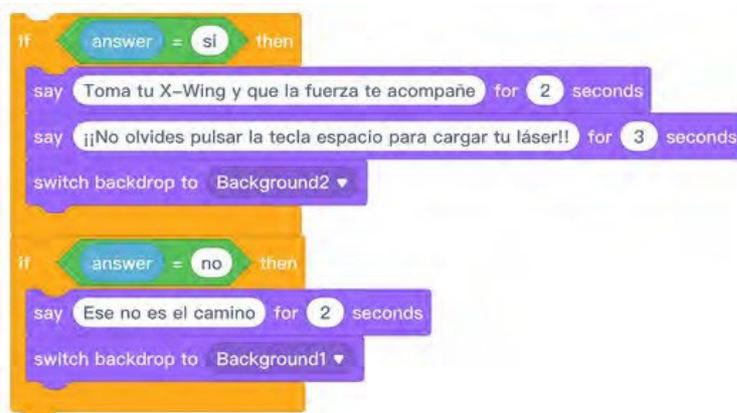
Ahora enlazaremos estos bloques con los anteriores



ACTIVIDAD 6



Por último, sólo queda unirse a la nueva respuesta del jugador. Si la respuesta es "sí" el juego comenzará y si la respuesta es "no" el juego volverá al principio. Para ello sólo tenemos que utilizar dos condicionales de "Control".



Estos nuevos bloques se unirán a los anteriores. El

resultado final debería ser algo parecido a esto



Lugar de respuesta



La respuesta se toma con el nombre del jugador, y se le pregunta si desea continuar el juego. Si la respuesta es "sí", se le dará la bienvenida y se le pedirá que pulse la tecla "espacio" para poder jugar. Esto es muy importante porque pasar la tecla "espacio" es el comando que permite iniciar el juego.

ACTIVIDAD 6

Por último, sólo queda dotar al X-Wing de un X-Wing en movimiento. Lo primero que haremos será llamar al objeto cuando el fondo cambie a "Fondo2" y decirle que aparezca con el traje de X-Wing, si no hacemos esto el objeto puede aparecer con el traje de "explosión".

```
when backdrop switches to Background2
  switch costume to X-Wing
  show
  go to x: -7 y: -152
  set size to 10%
```

Llamamos al objeto, le decimos que se muestre en el traje correspondiente, le fijamos un tamaño y le damos unas coordenadas X e Y.

Para dar movimiento a la nave, utilice las teclas de flecha izquierda y derecha. Es importante colocar los condicionales en una estructura "para siempre".

```
forever
  if key right arrow pressed? then
    change x by 10
  if key left arrow pressed? then
    change x by -10
  if on edge, bounce
```

Al cambiar X en 10 unidades estamos dando velocidad de desplazamiento, por lo que podemos adaptar la velocidad modificando el valor de X

Este sprite sólo se mostrará en el fondo donde se desarrolla el juego, que es "Fondo2", por lo que será necesario indicar que en el resto de fondos el objeto estará oculto.

```
when backdrop switches to Game Over
  hide

when backdrop switches to Victory
  hide
```

ACTIVIDAD 6

Paso 3. Crear clones. Creación de clones.

Para crear los "invasores" necesitamos subir el sprite "storm_troopers".



Este sprite debe estar oculto en todos los fondos excepto en el que tiene lugar el juego "Fondo2".



La función de este sprite será defender el objeto "Deatd_star" y atacar las defensas y la nave Ala-X. Para programar este sprite y darle movimiento, se crearán clones del sprite. El proceso de creación de clones se iniciará al pulsar la tecla "espacio". Pulsando esta tecla cambiamos al fondo del juego "Background2", establecemos el tamaño del sprite y le damos una posición en los ejes X e Y.



Ahora podemos empezar a crear los clones. Primero le diremos que cree cuatro filas de sprites a la izquierda del fondo, y luego le diremos que cree 10 clones en cada una de las filas cambiando el eje X en 35 unidades para crearlos a la derecha y el eje Y en -35 para crearlos hacia abajo para completar las cuatro filas.

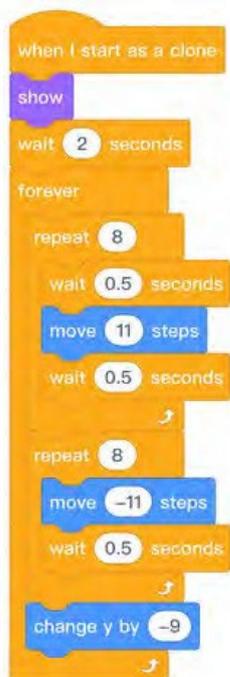
Todo debe reunirse en un único bloque de programación. El resultado debería ser el siguiente

ACTIVIDAD 6



Una variante del juego permitiría crear cuatro filas independientes de clones y programarlas para que su movimiento esté mejor coordinado, pero esto complica el juego. Esta variante se incluye en los archivos adjuntos.

Una vez creados los clones, se les dará movimiento.



1º desplazarse a la derecha y esperar 2º desplazarse a la izquierda y esperar 3º desplazarse hacia abajo

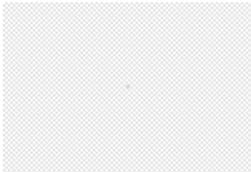
El sistema queda mejor ordenado si en lugar de un objeto se suben cuatro sprites y se crean los clones en cuatro filas, pero para ello hay que programar cada sprite por separado. Esa será la variante del juego que se incorporará a los archivos.

Paso 4. Creación de las defensas.



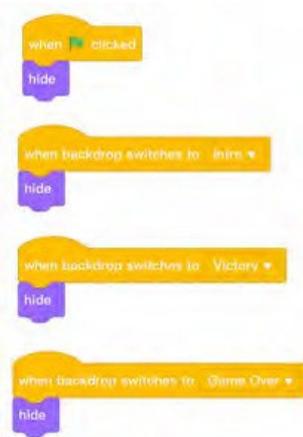
La creación de las defensas requiere subir el sprite "Tatooine_Building" y añadirle 6 disfraces. Primero subiremos el objeto y desde la opción "Disfraces" haremos una copia del sprite duplicándolo. A cada copia le borraremos parte del dibujo con el editor de disfraces para simular diferentes fases de destrucción.

ACTIVIDAD 6



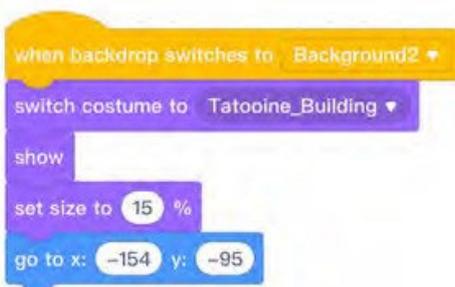
Es muy importante que el último traje del sprite sea una imagen totalmente vacía.

Esta operación debe repetirse tres veces o tantas veces como defensas queramos incluir en el juego.



Para programar las defensas, comience por indicar que no deben mostrarse sobre fondos distintos de "Fondo2".

Por último, sólo queda indicar que aparecen en el fondo del juego y darles una ubicación.



Es muy importante indicar que al principio se cambie al primer traje donde se muestra el dibujo completo. Luego se fija un tamaño y coordenadas en la parte inferior de la pantalla dejando el espacio necesario para el objeto X-Win detrás de las defensas. Este programa se arrastra a cada una de las defensas cambiando sólo el valor del eje X para que queden separadas entre sí a la misma distancia. En una siguiente fase del programa veremos que ocurre cuando un láser incide sobre las defensas. El resultado será el siguiente.

ACTIVIDAD 6

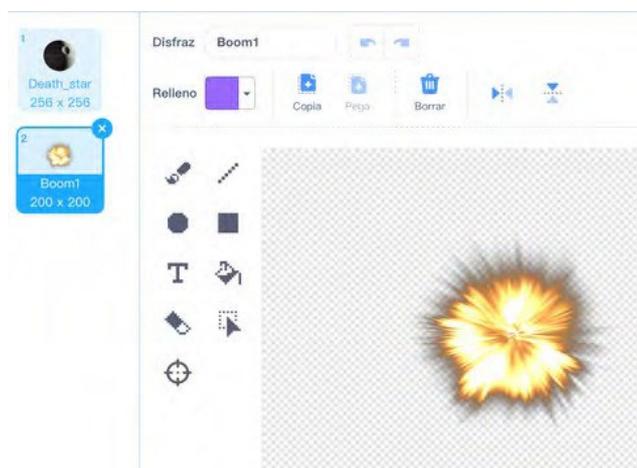


Paso 5. Creación de la nave atacante (Estrella de la Muerte).

Para crear la nave atacante subiremos el sprite Death_star



A este sprite le añadiremos un disfraz que será una explosión, que puede ser tomada del repertorio de sprites que ofrece Mblock o subir nuestro propio sprite de explosión.



ACTIVIDAD 6

De nuevo, indicaremos que el sprite se oculte en todos los fondos excepto en el fondo de juego "Fondo2" y le daremos movimiento. La función de este sprite es moverse de izquierda a derecha y de derecha a izquierda disparando un láser.

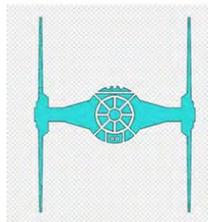


Es muy importante indicar que el traje en el que se muestra es "Death_star" para evitar que aparezca como una explosión.

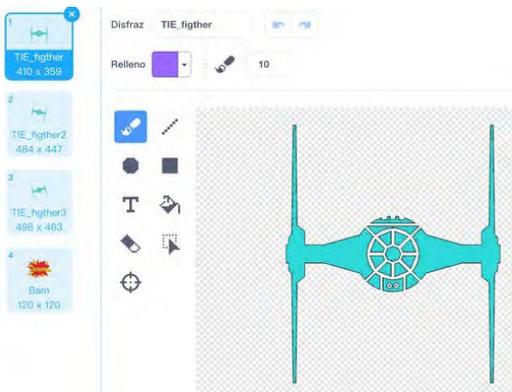
Establecemos un tamaño y unas coordenadas en los ejes X e Y. Por último, dentro de una estructura "forever", indicamos el movimiento a derecha e izquierda deslizando el sprite.

Paso 6. Creación de la nave secundaria (TIE-fighter) Creación de la nave secundaria (TIE-fighter)

Debemos cargar el sprite "TIE_fighter".



Se han añadido tres trajes más a este sprite. Dos de ellos son los sprites inclinados hacia la izquierda y hacia la derecha y un disfraz más de explosión.



Este sprite saldrá de la Estrella de la Muerte y tendrá movimientos aleatorios y finalmente apuntará a la nave X-Wing y chocará contra ella. El objetivo del juego será derribar este sprite o evitar chocar contra él.

ACTIVIDAD 6

Empezaremos diciendo que el sprite está oculto en todos los fondos y que se presenta con el traje "TIE-fighter".



Para que el sprite aparezca en el juego es necesario crear una variable que llamaremos "tiempo". Desde la sección "Variables", haz clic en "Crear una variable", nómbrala "tiempo" y OK.



Al pulsar la "bandera verde", el tiempo se pondrá a 0 y se creará un temporizador cuando se pulse la tecla de espacio.



Para crear el temporizador, dentro de una estructura "forever" indicaremos esperar un segundo y cambiar el valor de la variable tiempo a 1

Paso 7. Interacción de los sprites

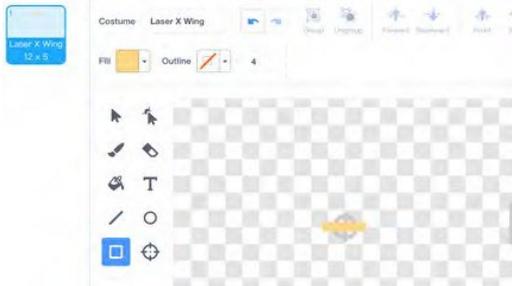
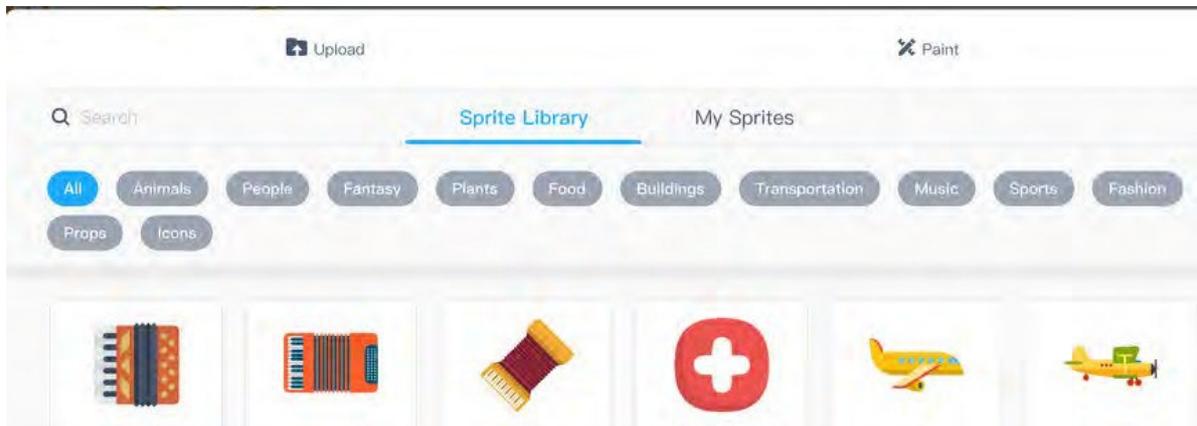
En esta sección vamos a ver cómo interactúan los sprites para el desarrollo del juego.

Necesitamos crear dos sprites que actuarán como láseres, uno para la nave atacante (Death-stat) y otro para la nave X-Wing.

Crear láseres para X-Wing y para Death_star

En la sección Objetos, haga clic en "Añadir" y luego en "Pintar".

ACTIVIDAD 6



A partir de esta pantalla dibujaremos un rectángulo con un color que le permita destacar sobre un fondo negro. Es importante situar el objeto en el centro. Lo llamaremos "Ala Láser X". Realizaremos la misma operación para crear un segundo sprite de otro color al que llamaremos "Láser Estrella de la Muerte".

Los dos sprites deben estar ocultos en todos los fondos del juego y se mostrarán al pulsar la tecla "espacio".

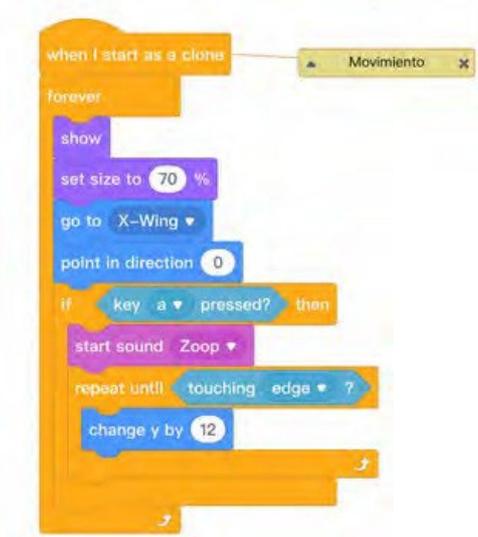
Láser X-Wing.

Este láser se dispara desde la nave X-Wing y te permitirá eliminar a los clones (Storm-troopers), la nave Estrella de la Muerte, la nave auxiliar TIE_Fighter y las defensas. También te permitirá sumar puntos por cada clon de Storm_troopers eliminado.



Es necesario crear una variable que llamaremos "PUNTOS" que pondremos a 0 y un clon de este objeto.

ACTIVIDAD 6



Para dar movimiento al sprite lo uniremos a la nave X-Wing para que el disparo siempre salga de la nave al pulsar la tecla 'a'. El movimiento se crea cambiando el eje Y por 12 unidades y se repite hasta que el objeto toca un borde.

Para interactuar con las defensas utilizaremos el bloque de programación "enviar un mensaje". La estrategia es que cuando toque una defensa envíe el mensaje "stop" (detener) y cuando reciba el mensaje se esconda. Es necesario programar qué ocurre cuando toca cada una de las defensas.



Láser Estrella_de_la_Muerte

Al principio de la programación indicaremos que el sprite estará oculto en todos los fondos del juego. Este sprite se dispara desde la nave Death_star, por lo que tendremos que fijar el sprite a la nave. Además, para hacer el juego más interesante y que la frecuencia de disparo no sea siempre la misma, indicaremos que el disparo sea aleatorio entre un número aleatorio entre 1 y 3. Esta frecuencia se puede modificar ampliando el rango entre los números. Añadiremos una velocidad modificando el eje Y en -8 unidades y este movimiento descendente se repite hasta que toca una de las defensas. Cuando eso ocurra se esconde

ACTIVIDAD 6



Indicamos que el objeto debe estar unido al objeto "Estrella de la Muerte" para que los disparos se realicen siempre desde la nave. Aquí también debemos indicar qué ocurre cuando el objeto choca contra las defensas. El objeto se mueve con un valor del eje Y de -8 hasta que choca con las defensas y se esconde.

Clones Storm_troopers

Estos sprites se mueven hacia los lados y hacia abajo. Su función es defender la nave Estrella_de_la_Muerte, atravesar las defensas y, si alcanzan una determinada posición, terminar la partida derrotando al jugador. La derrota también se produce si alguno de los clones toca la nave X-Wing.

Cuando uno de estos sprites toque el láser del X-Wing sumará un punto al jugador. Es importante que el sprite del láser se detenga cuando un clon es alcanzado para evitar que continúe su trayectoria ascendente y elimine más clones.

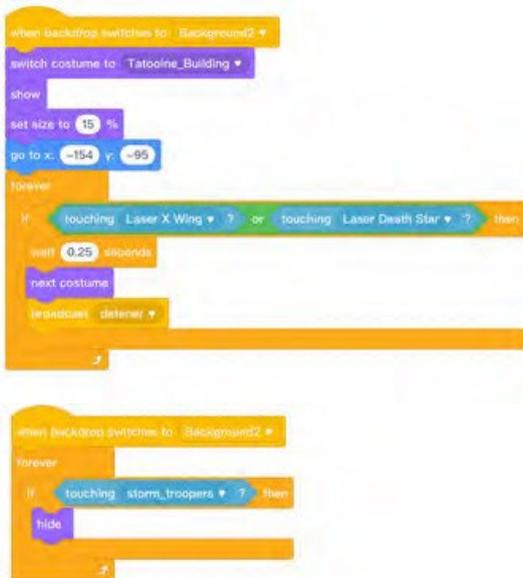
ACTIVIDAD 6



Cuando golpea el objeto Laser X Wing suma un punto

Cuando alcance una posición en el eje Y (-152) el fondo cambiará a Game Over y el juego se detendrá.

Defensas Tattoine_Construcción



Programaremos una defensa y luego copiaremos el programa en las otras dos.

Al principio diremos que cuando pulsemos la bandera verde y en los fondos Enter, Game Over y Victory se ocultará.

Las defensas se destruyen cuando tocan alguno de los objetos "Láser". Cada vez que un láser toca la defensa cambia de disfraz hasta llegar al último, que es una imagen vacía

Si la defensa toca un clon de Storm_troopers, la defensa se oculta directamente..

X-Nave de ala

La nave X-Wing será eliminada y el juego terminará cuando:

- Recibe tres impactos láser del láser de la nave Estrella de la muerte.
- Toca cualquiera de los clones de Storm Troopers.
- Golpea la nave TIE-Fighter..

ACTIVIDAD 6

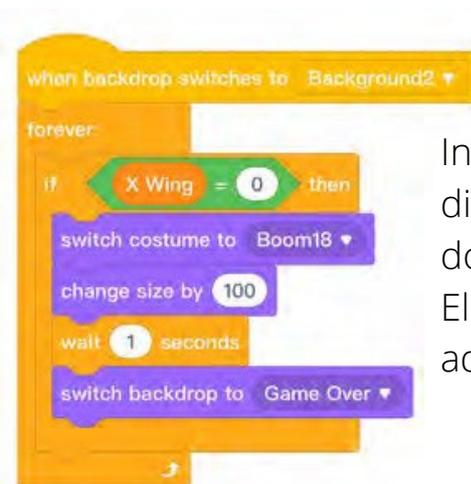
Cada una de estas opciones debe programarse por separado. Para programar la opción A) debemos crear una variable en la que se almacenarán los impactos recibidos. Esta variable se llamará "Ala X" y su valor será 3

Los impactos láser de la Estrella de la Muerte se programarán a partir del sprite "Estrella de la Muerte láser".



Lo que indicamos es que si el láser impacta en el Ala X cambiará la variable a -1. Para evitar que una vez que el sprite siga restando puntos mientras se mueve, decimos que espere 0.25 segundos. Este clon se ocultará cuando toque un borde.

Lo que queda ahora es indicar lo que ocurre cuando la variable X Wing tiene un valor de 0 puntos..



Indicamos que si el valor de la variable = 0 cambie el disfraz de explosión y cambie el fondo a Game Over donde termina el juego.

El cambio de tamaño del disfraz es necesario para adaptarlo al juego.

ACTIVIDAD 6

La opción B) debe programarse en paralelo al final del juego. El juego termina cuando se alcanza una puntuación de 40 puntos, que es el número de clones que aparecen al principio del juego.



Tienes que cargar un sonido del tema Star Wars Intro para que suene cuando alcances los 40 puntos. Si, por el contrario, la nave Ala X es alcanzada por un clon, el fondo cambiará a Game Over y la partida terminará.

Esta segunda parte es importante ponerla en paralelo porque queremos que los dos programas se ejecuten al mismo tiempo y no uno detrás de otro.



Lo que decimos en esta parte del programa es que cuando se alcancen los 40 puntos, se cambie el fondo a "Victoria" y se esconda el barco.

Podemos utilizar estos dos programas para indicar lo que ocurre en la opción B), cuando un clon de Storm Troopers golpea la nave X Wing y así salvar los programas. Finalmente la situación quedaría así:



ACTIVIDAD 6

En la opción C) el programa es similar



Sólo indicamos que el juego termina cuando el TIE Fighter impacta con Ala X.

Nave de combate TIE

Para programar esta nave, se creó una variable llamada "tiempo" para determinar cuándo empezaría a actuar esta nave. En este caso le hemos asignado un valor de 10 segundos. La nave se fijará en el sprite "Estrella de la Muerte" y se mostrará, esperará 0,25 segundos y comenzará su movimiento. Lo que queremos es que la nave muestre una serie de movimientos aleatorios antes de dirigirse hacia la nave Ala X para golpearla. El objetivo del juego será evitar el impacto o derribarla con un disparo láser.



Estos son los movimientos aleatorios. De esta forma, cada vez que se inicie el juego, la nave realizará movimientos impredecibles. Cuando haya terminado de hacer estos movimientos, se dirigirá hacia la nave Ala X.

ACTIVIDAD 6



En este apartado indicaremos que si la nave toca un láser Ala X cambiará su disfraz a una explosión y se esconderá.

En el caso de que evitemos el impacto y no derribemos la nave, ésta se ocultará 20 segundos después de iniciarse el movimiento, impidiendo así que la Neve permanezca en el juego.

Es importante asegurarse de que la nave permanece oculta en todos los fondos del juego y sólo se muestra cuando se le indica.

Nave Estrella de la Muerte



Cuando subimos el objeto "Estrella de la Muerte" ya indicamos su movimiento, lo que haremos ahora es programar su eliminación del juego. Lo que haremos será crear una variable llamada "Estrella de la Muerte" cuya función será almacenar los impactos que reciba. Se programará para que la nave sea eliminada cuando reciba tres impactos.

Es importante que pongamos el valor de la variable a 0 cuando creamos el barco.

ACTIVIDAD 6

```
when backdrop switches to Background2
  forever
    if touching Laser X Wing? then
      wait 0.25 seconds
      change Death Star by 1
```

Indicamos que si al tocar el láser cambia el valor de la variable en 1

```
when backdrop switches to Background2
  forever
    if Death Star = 3 then
      switch costume to Boom1
      wait 1 seconds
      change size by 40
      wait 1 seconds
      hide
```

Indicamos que cuando la variable tiene un valor de 3 cambia su disfraz a explosión, espera un segundo y se esconde.

```
when I start as a clone
  forever
    if Death Star = 3 then
      hide
```

Es muy importante volver a la programación del objeto "Láser de la Estrella de la Muerte" e indicar que cuando la nave es eliminada el láser deja de disparar.

Sprite "Victoria

```
when clicked
  hide

when backdrop switches to Intro
  hide

when backdrop switches to Background2
  hide

when backdrop switches to backdrop1
  hide

when backdrop switches to Game Over
  hide
```

Se trata de un objeto opcional que se coloca al final del juego cuando has alcanzado los 40 puntos. Es un objeto gif con 33 disfraces. El objeto ha sido recogido de la web y al ser un objeto opcional se puede evitar. Su programación es la siguiente:

Lo ocultamos en todos los fondos del juego y sólo cuando el fondo cambia al fondo "Victoria" se muestra y se pide cambiar de traje.

```
when backdrop switches to Victory
  show
  set size to 125%
  next costume
```

ACTIVIDAD 6



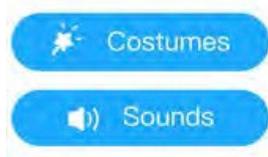
Nota muy

Todas las variables que se han incluido en el juego deben estar ocultas excepto la que indica los puntos que estamos alcanzando.

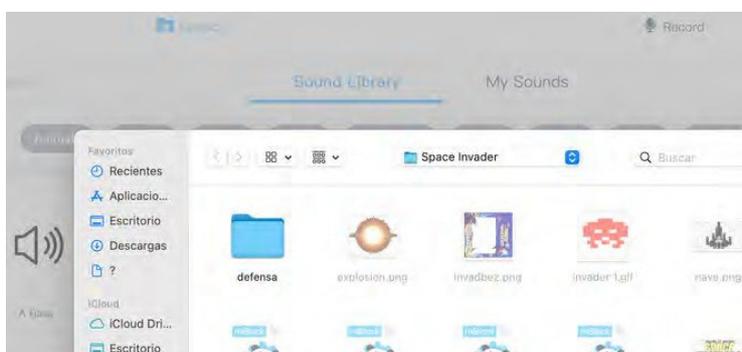
Podemos seleccionar que se muestren las variables pero quitan visibilidad al juego.

Paso 8. Incorporar sonidos

Para añadir sonidos al videojuego, sólo tienes que seleccionar "Sonidos" en la pestaña "Sprites".



A continuación, selecciona "Añadir sonido".



Y hacemos clic en "Cargar". Accedemos a nuestras carpetas en el ordenador y seleccionamos los sonidos que queremos incorporar al juego.

ACTIVIDAD 6



En nuestro juego estos son los sonidos que se incorporan

A partir del sprite "X Wing" podemos incorporar sonidos



ACTIVIDAD 6

RECURSOS



EVALUACIÓN DEL ALUMNO

A evaluar:

- Uso básico de Mblock Carga
 - de sprites Creación de
 - fondos
 - Creación de sprites con trajes
 - Mostrar y ocultar sprites en la secuencia adecuada
 - Movimientos básicos con sprites
 - Utilizar estructuras condicionales
- Diseñar una estrategia global del
- proyecto Dividir el proyecto en tareas más sencillas
- Revisar versiones de proyectos y corregir posibles errores
- Crear clones de sprites
- Trabajar con variables

BIBLIOGRAFÍA

Marjil, Made (2014). Aprende a programar con Scratch: Una introducción visual a la programación con juegos, arte, ciencia y matemáticas.

Morris, Mike (2019). Programación con Scratch: Un tutorial en profundidad sobre programación con Scratch para principiantes: 1

Sweigart, Al (2021). Scratch 3 Programming Playground: Aprende a programar creando juegos geniales

Yuqiang, Liao & Tongzheng; Zhao.

Mblock Kid maker. Rocas con los robots (PDF)



Tutoriales



ADAPTACIÓN A OTROS NIVELES

El videojuego permite sustituir el movimiento de la nave X Wing mediante teclas por un Joystick que también puede utilizarse como disparador. Se pueden añadir LEDs para indicar los impactos que ha recibido la nave X Wing. Se puede incluir un botón para disparar los láseres del Ala X.

MÁS INFORMACIÓN

Enlace a las carpetas con los objetos para diseñar el software y los juegos



ACTIVIDAD 7

TÍTULO Mbot Racing

RESUMEN

Los concursos demuestran el éxito. Premiamos el éxito con estos concursos.

En la vida escolar, las competiciones tienen una importancia especial para los alumnos. Porque los instructores pueden utilizar las competiciones para aumentar la permanencia de sus lecciones.

En este proyecto, queríamos divertirnos y codificar nuestros Mbots.

Nuestro proyecto se basa en que los alumnos responden a las preguntas que previamente se han ordenado con los códigos de sus móviles, y el Mbot que utilizan por cada respuesta correcta avanza un poco. El primero que llega a la meta por Mbots ganar la competencia.

AUTOR/S

FECHA 24/02/2022

VERSIÓN 1

Sultantepe Prof. Dr. Cemil Taşcıoğlu Ortaokulu

OBJETIVOS DIDÁCTICOS

Con este estudio, el alumno;

- Saber escribir código en el programa Mblock.
- Saber utilizar el programa Kahoot.
- Saber utilizar el mbot controlado por Bluetooth.
- Aprender divirtiéndose.
- Desarrollar la cultura general.

ACTIVIDAD 7

- CIENCIA
- TECNOLOGÍA
- MATEMÁTICAS
- GEOGRAFÍA/HISTORIA

- IDIOMAS
- LITERATURA
- MÚSICA
- OTROS

NIVEL DE ESTUDIOS

Esta actividad está preparada para...

- 12 - 14 AÑOS
- 14 - 16 AÑOS
- OTROS

HERRAMIENTAS NECESARIAS

Lista de material necesario

- Ordenador
- Mblock versión en
- línea Programa Kahoot
- Dispositivos móviles
- Bluetooth Mbot
- Pegamento, cartón, tijeras
- Pintura, pincel

ACTIVIDAD 7

DESARROLLO DE LA ACTIVIDAD

Paso 1: Creación de la plataforma de competición



Paso 2: Prepararse para competir con Kahoot

En este proyecto, nuestro objetivo no es sólo estudiar robótica, sino también alcanzar los logros de la asignatura. Por este motivo, nuestros alumnos se preparan para la competición haciendo pruebas previas en Kahoot para conocer las preguntas que se les harán.



ACTIVIDAD 7

Paso 3. Las preguntas se preparan en el programa Mblock.



RECURSOS



ACTIVIDAD 7



EVALUACIÓN DEL ALUMNO

Serán evaluados:

- Puntuación del alumno en Kahoot.
- Tiempo de respuesta de los estudiantes a las preguntas del bloque Mblock
- Número de respuestas correctas dadas por el alumno a las preguntas del bloque Mblock
- El robot puede seguir una línea recta, una curva a la derecha o una curva a la izquierda.

BIBLIOGRAFÍA

https://www.makeblock.es/productos/robot_educativo_mbot/

<https://kahoot.com/>

ACTIVIDAD 7

ADAPTACIÓN A OTROS NIVELES

Dado que el estudio se realizó en el marco del proyecto, se preparó una plataforma breve. Se puede crear una plataforma más larga.

Se pueden hacer competiciones más divertidas con más Mbots.

MÁS INFORMACIÓN

En el proyecto no se han utilizado sensores distintos de Bluetooth. Quienes deseen contribuir al proyecto

1. Utilizando los LED del Mbot, se puede garantizar que éste responda a las respuestas correctas e incorrectas con colores verde/rojo.
2. Mbot se moverá más recto si se apoya con códigos de seguimiento de línea y sensor
3. En caso de que el Mbot llegue al punto final, se puede asegurar que reproducirá la música determinada por los códigos.
4. Colocando un sensor de distancia ultrasónico delante del Mbot, se puede evitar que golpee la pared/cartón cuando llegue al punto final.

ACTIVIDAD 8

TÍTULO Construcción y programación de un robot de seguimiento de líneas

RESUMEN

La tecnología es cada vez más necesaria en nuestras vidas, y es de esperar que lo sea aún más en el futuro. Por ello, es necesario que nuestros alumnos empiecen a conocerla desde una edad temprana.

El objetivo de esta actividad es que los alumnos diseñen, construyan y programen un robot seguidor de líneas, construyan su propio circuito y, por último, personalicen el robot.

AUTOR/S

IES MEDITERRÁNEO

FECHA 10/02/2022

VERSIÓN 1

OBJETIVOS DIDÁCTICOS

Los objetivos son los siguientes

- Familiarizarse con el diseño en 3-D y todas las posibilidades que conlleva.
- Diseñar objetos en 3D.
- Para imprimir objetos en 3D.
- Fomentar el aprendizaje interdisciplinar y el trabajo en equipo.
- Mejorar la motivación y el interés de los alumnos.
- Mejorar la visión espacial.
- Aumentar la creatividad.
- Planificar un proyecto desde el principio.
- Mejorar la capacidad de resolución de problemas.
- Instalación y programación de componentes electrónicos

ACTIVIDAD 8

CIENCIA TECNOLOGÍA

MATEMÁTICAS

GEOGRAFÍA/

HISTORIA

IDIOMAS

LITERATURA

MÚSICA

OTROS

NIVEL DE ESTUDIOS

Esta actividad está preparada para...

12 - 14 AÑOS

14 - 16 AÑOS

OTROS

HERRAMIENTAS NECESARIAS

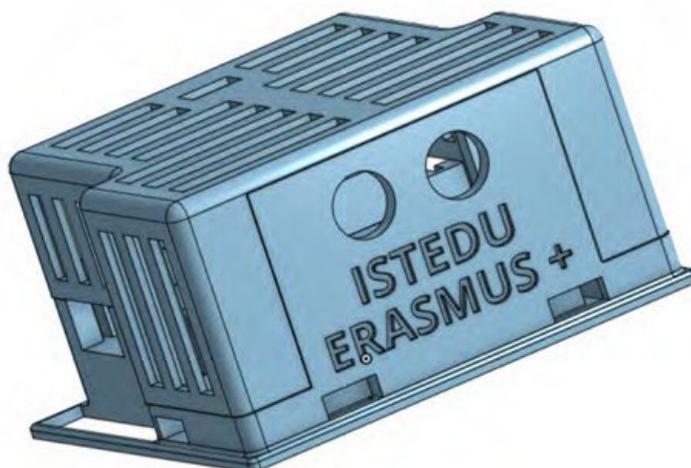
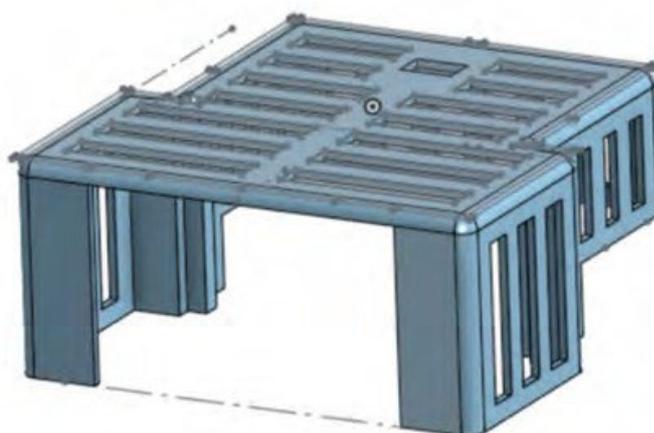
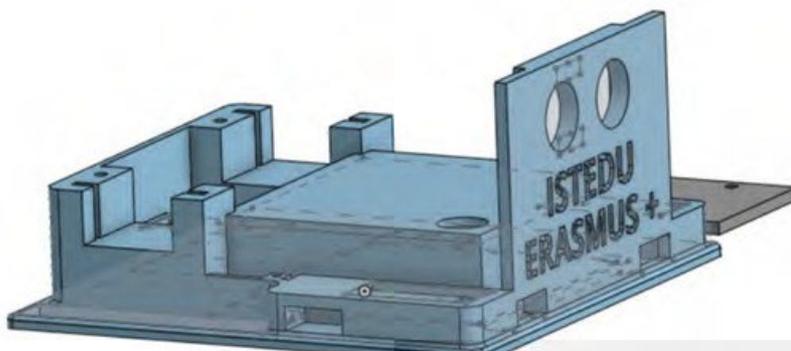
Se necesitan los siguientes materiales:

- Ordenador.
- Mblock versión 3.11
- Software de diseño 3D
- Impresora 3D
- Placa Arduino
- Shield para placa Arduino Servo 360
- sg90
- Sensor de distancia HC-SR04
- Sensor de línea TCRT5000
- Ruedas
- Cables F-F
- Portapilas

ACTIVIDAD 8

DESARROLLO DE LA ACTIVIDAD

Paso 1: Impresión 3D de las piezas



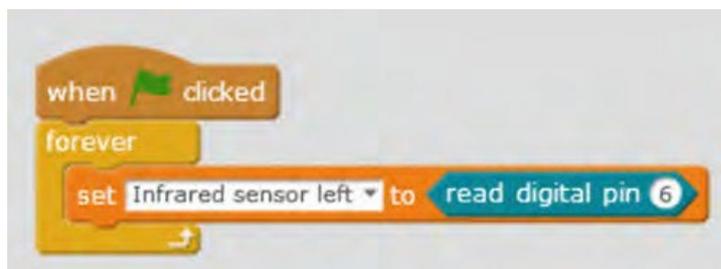
ACTIVIDAD 8

Paso 2. Conectar todos los componentes Conecte todos los componentes siguiendo el diagrama siguiente:

- Motor derecho al pin digital número 9
- Motor izquierdo al pin digital número 8
- Sensor infrarrojo izquierdo a pin digital número 7
- Sensor infrarrojo derecho a pin digital número 6
- Sensor ultrasónico o sensor de distancia:
- Trig al pin digital número 13 y Echo al pin digital número 12.

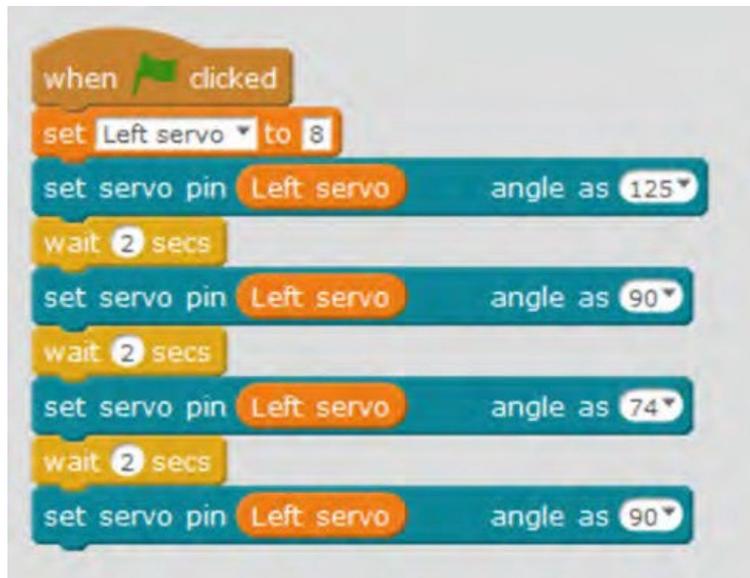
Paso 3. Una vez conectados todos los componentes, compruebe uno a uno que funcionan correctamente.

- Sensores infrarrojos:
 - Abrimos MBlock e introducimos el siguiente programa en modo Mblock.



- Colocamos el sensor, que vamos a probar, sobre una hoja en blanco con líneas negras. Para que funcione correctamente, debe dar 0 cuando está sobre blanco y 1 cuando está sobre negro.
 - Hacemos lo mismo con el otro sensor.
- Servos:
 - Abrimos Mblock y cargamos el siguiente programa:

ACTIVIDAD 8



- El motor izquierdo debe girar en una dirección durante 2 segundos, detenerse durante otros 2 segundos, girar en la dirección opuesta y finalmente detenerse.
- Lo mismo debe hacerse con el otro motor.

Paso 4. Imprime las hojas de circuitos y pégalas en el suelo o en una superficie rígida.

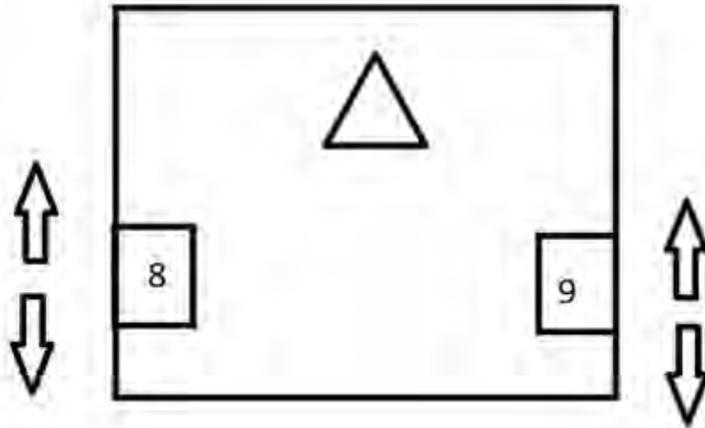
Paso 5 Programe el robot para que siga la línea.

No es una tarea difícil, pero si se intenta abordar sin dividirla en diferentes partes puede resultar complicado.

1. Programación de movimientos

- Prográmalo para que camine recto. Para ello, hay que ajustar los ángulos de giro de los dos servos. Utilizando un diagrama similar al siguiente, indica el ángulo de giro para que se mueva en cada sentido. Es conveniente ajustar los ángulos para diferentes velocidades.

ACTIVIDAD 8

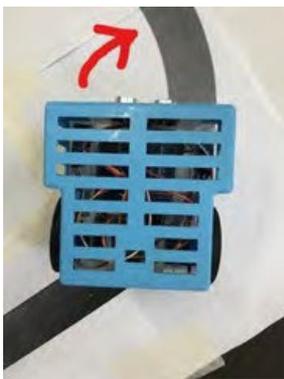


- Prográmalo para que gire a la derecha, tienes que ajustar los ángulos de los dos servos, sería conveniente probar diferentes velocidades.
- Prográmalo para que gire a la izquierda, hay que ajustar los ángulos de los dos servos, sería conveniente probar diferentes velocidades.

2. Seguir parte de la línea

Ahora debemos programar conjuntamente los sensores infrarrojos y los movimientos del robot.

Cuando el robot está en el circuito, pueden darse al menos tres situaciones:



El sensor izquierdo está en la línea negra y el sensor derecho está en la línea blanca, el robot debe girar a la derecha..

```
Arduino Program
set Left servo to 3
set Right servo to 9
for ever
  set Infrared sensor right to read digital pin 7
  set Infrared sensor left to read digital pin 6
  if Infrared sensor left = 0 and Infrared sensor right = 1 then
    set servo pin Left servo angle as 125°
    set servo pin Right servo angle as 90°
  if Infrared sensor left = 0 and Infrared sensor right = 0 then
    set servo pin Left servo angle as 125°
    set servo pin Right servo angle as 90°
  wait 0.5 secs
```

Es posible que siga hacia el interior de la curva, para volver al interior debe seguir girando a la derecha.

ACTIVIDAD 8



```
Arduino Program
set Left servo to 8
set Right servo to 9
forever
  set Infrared sensor right to read digital pin 7
  set Infrared sensor left to read digital pin 6
  if Infrared sensor left = 0 and Infrared sensor right = 0 then
    set servo pin Right servo angle as 125°
    set servo pin Left servo angle as 74°
```



```
Arduino Program
set Left servo to 8
set Right servo to 9
forever
  set Infrared sensor right to read digital pin 7
  set Infrared sensor left to read digital pin 6
  if Infrared sensor left = 1 and Infrared sensor right = 0 then
    set servo pin Left servo angle as 90°
    set servo pin Right servo angle as 69°
  if Infrared sensor left = 0 and Infrared sensor right = 0 then
    set servo pin Left servo angle as 90°
    set servo pin Right servo angle as 69°
  wait 0.5 secs
```

Es aconsejable probar cada una de estas situaciones por separado, para saber si cada una de ellas se ha programado correctamente. Seguir todo el circuito.

3. En este caso, hay que programar conjuntamente las situaciones del apartado anterior, intentando hacer el circuito despacio y aumentando la velocidad para que se haga lo más rápido posible.

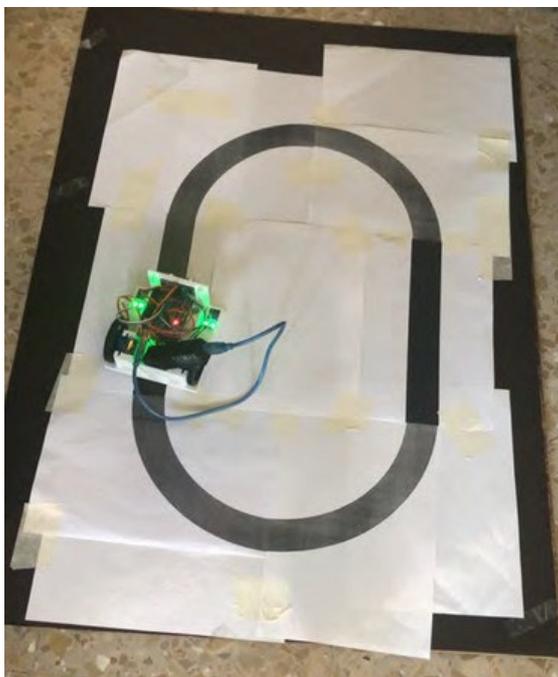
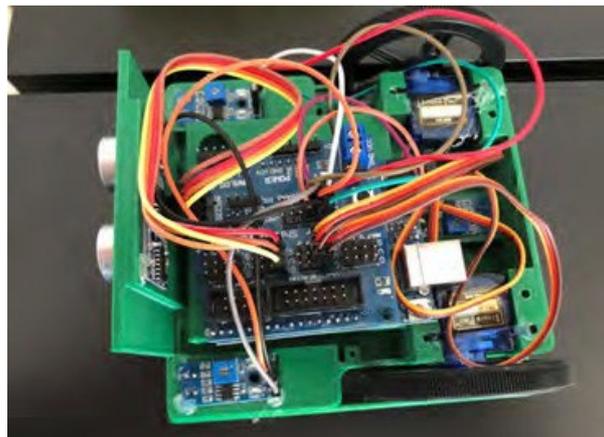
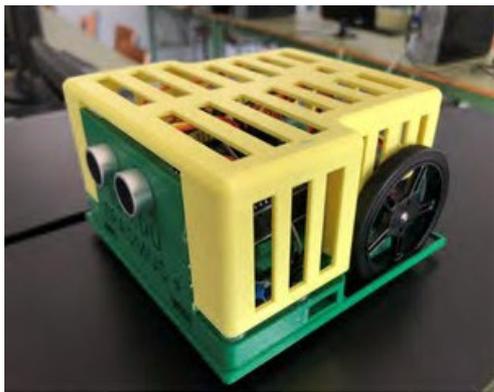
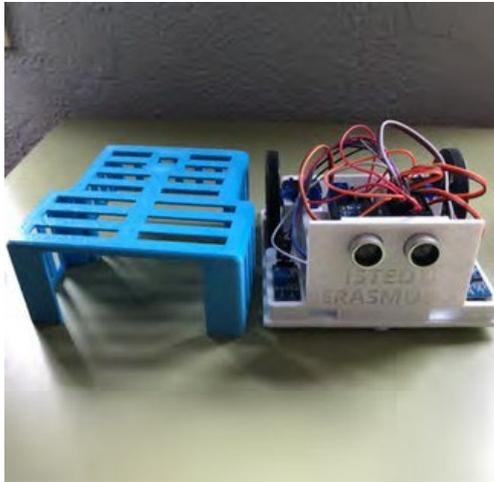
ACTIVIDAD 8

```
Arduino Program
set Left servo* to 8
set Right servo* to 9
forever
  set Infrared sensor right to read digital pin 7
  set Infrared sensor left to read digital pin 6
  if Infrared sensor left = 0 and Infrared sensor right = 0 then
    set servo pin Right servo angle as 125
    set servo pin Left servo angle as 75
  if Infrared sensor left = 1 and Infrared sensor right = 0 then
    set servo pin Left servo angle as 90
    set servo pin Right servo angle as 65
  if Infrared sensor left = 0 and Infrared sensor right = 0 then
    set servo pin Left servo angle as 90
    set servo pin Right servo angle as 65
    wait 0.5 secs
  if Infrared sensor left = 0 and Infrared sensor right = 1 then
    set servo pin Left servo angle as 125
    set servo pin Right servo angle as 90
  if Infrared sensor left = 0 and Infrared sensor right = 0 then
    set servo pin Left servo angle as 125
    set servo pin Right servo angle as 90
    wait 0.5 secs
```

RECURSOS



ACTIVIDAD 8



EVALUACIÓN DEL ALUMNO

Se evaluará:

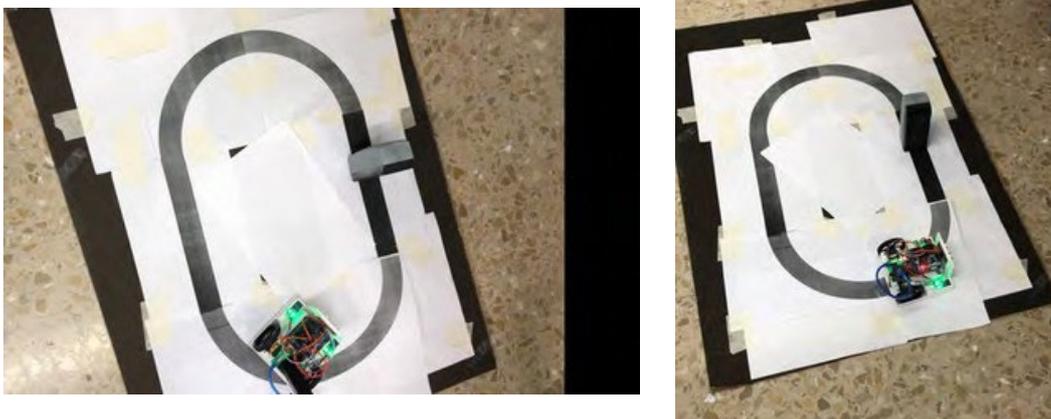
- Impresión de las diferentes partes del robot.
- Instalación de los componentes electrónicos.
- Funcionamiento correcto de todos los componentes electrónicos mediante la programación individual de cada uno de ellos.
- Programación de los diferentes movimientos del robot.
- El robot puede seguir una línea recta, una curva a la derecha o una curva a la izquierda. El robot puede hacer todo el circuito sin salirse.
- El robot puede hacer el circuito lo más rápido posible.

BIBLIOGRAFÍA

<https://www.arduino.cc/>

ADAPTACIÓN A OTROS NIVELES

Se pueden añadir obstáculos por el camino para que los evites y vuelvas a la línea.



Se puede programar para que no salga del circuito, es decir, que cuando detecte la línea negra, gire hacia dentro. Otra opción es hacer un sumo, con gafas dentro del circuito y debe conducir fuera del circuito.

ACTIVIDAD 9

TÍTULO Construcción y programación de un robot seguidor de luz

RESUMEN

La tecnología es cada vez más necesaria en nuestras vidas, y es de esperar que lo sea aún más en el futuro. Por ello, es necesario que nuestros alumnos empiecen a conocerla desde una edad temprana.

El objetivo de esta actividad es que los alumnos diseñen, construyan y programen un robot seguidor de líneas, construyan su propio circuito y, por último, personalicen el robot.

AUTOR/S

IES MEDITERRÁNEO

FECHA 10/02/2022

VERSIÓN 1

OBJETIVOS DIDÁCTICOS

Los objetivos son los siguientes

- Familiarizarse con el diseño en 3-D y todas las posibilidades que conlleva.
- Diseñar objetos en 3D.
- Para imprimir objetos en 3D.
- Fomentar el aprendizaje interdisciplinar y el trabajo en equipo.
- Mejorar la motivación y el interés de los alumnos.
- Mejorar la visión espacial.
- Aumentar la creatividad.
- Planificar un proyecto desde el principio.
- Mejorar la capacidad de resolución de problemas.
- Instalación y programación de componentes electrónicos.

ACTIVIDAD 9

CIENCIA

TECNOLOGÍA

MATEMÁTICAS

GEOGRAFÍA/HISTORIA

IDIOMAS

LITERATURA

MÚSICA

OTROS

NIVEL DE ESTUDIOS

Esta actividad está preparada para...

12 - 14 AÑOS

14 - 16 AÑOS

OTROS

HERRAMIENTAS NECESARIAS

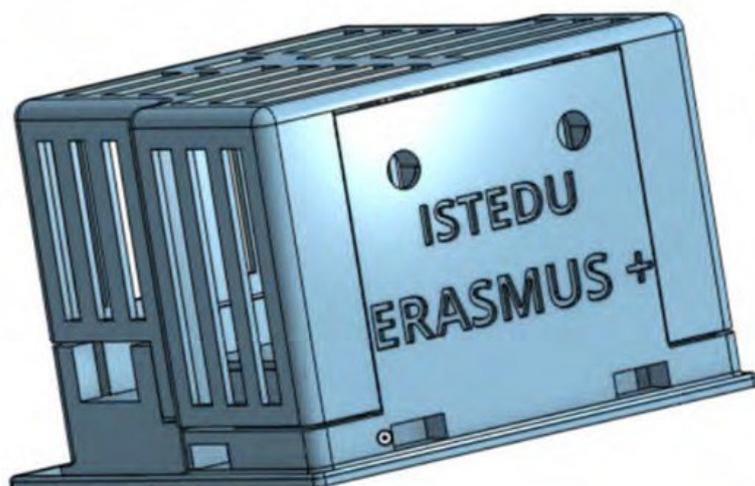
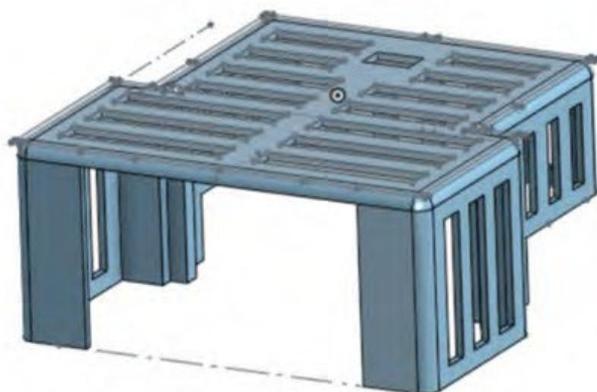
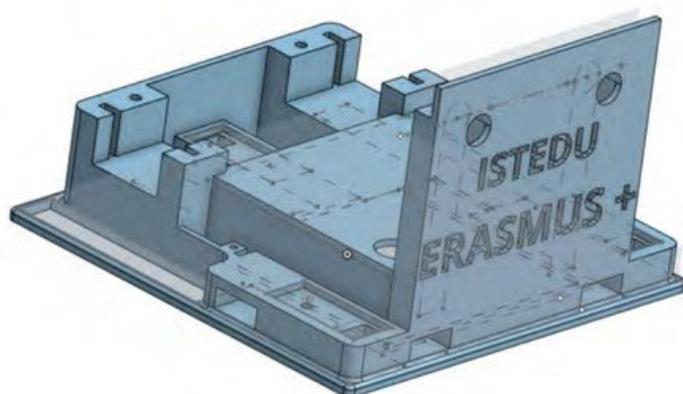
Se necesitan los siguientes materiales:

- Ordenador.
- Mblock versión 3.11
- Software de diseño 3D
- Impresora 3D
- Placa Arduino
- Shield para placa Arduino
- Servo 360 sg90
- Sensor de luz LDR
- Ruedas
- Resistencias
- Cables F-F
- Portapilas

ACTIVIDAD 9

DESARROLLO DE LA ACTIVIDAD

Paso 1: Diseño e impresión 3D de las piezas



ACTIVIDAD 9

Paso 2. Conectar todos los componentes Conecte todos los componentes siguiendo el diagrama siguiente:

- Motor derecho al pin digital número 9
- 9 Motor izquierdo al pin digital número 8
- Conecta los divisores de tensión de la izquierda a A0 y los de la derecha a A1.

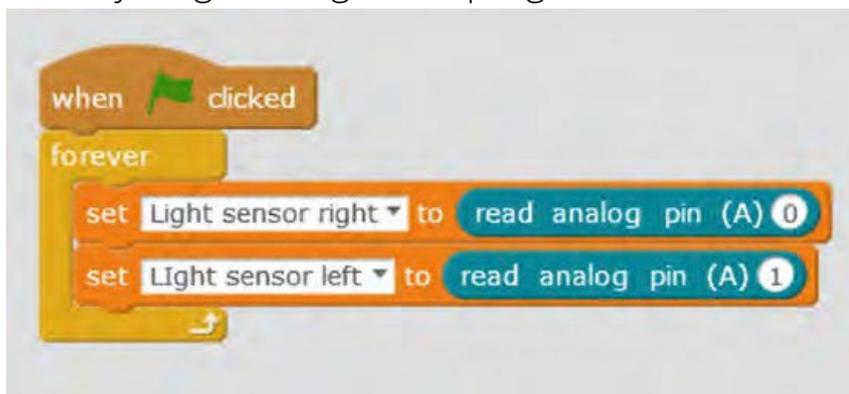
Paso 3. Una vez conectados todos los componentes, compruebe uno a uno que funcionan correctamente:

- Servos:
 - Abra Mblock y cargue el siguiente programa:



El motor izquierdo debe girar en una dirección durante 2 segundos, detenerse durante otros 2 segundos, girar en la dirección opuesta y finalmente detenerse.

- Lo mismo debe hacerse con el otro motor.
- Sensor de luz LDR:
 - Abra Mblock y cargue el siguiente programa :



ACTIVIDAD 9

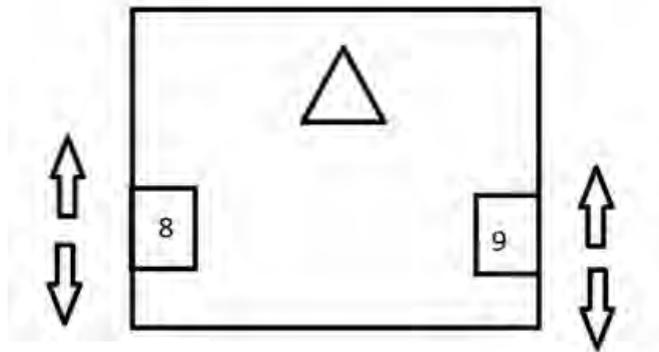
Si funcionan correctamente deberían dar un valor entre 0 y 1023, si ponemos la mano delante de los sensores o los iluminamos los valores deberían cambiar.

Paso 4. Programa el robot para que siga la luz.

No es una tarea difícil, pero si intentas abordarla sin dividirla en diferentes partes, puede resultar complicado. La dividiremos en 3 partes.

1. Movimientos del programa

- Prográmalo para que camine recto. Para ello, hay que ajustar los ángulos de giro de los dos servos. Utilizando un diagrama similar al siguiente, indica el ángulo de giro para que se mueva en cada sentido



Es conveniente ajustar los ángulos para diferentes velocidades.

- Prográmalo para que gire a la derecha, hay que ajustar los ángulos de los dos servos, sería recomendable probar diferentes velocidades.
- Prográmalo para que gire a la izquierda, hay que ajustar los ángulos de los dos servos, sería recomendable probar diferentes velocidades.

2. Luz de seguimiento

Ahora debemos programar conjuntamente los sensores de luz y los movimientos del robot.

La operación será la siguiente:

- Al principio debe detectar la luz ambiente y guardarla en una variable.

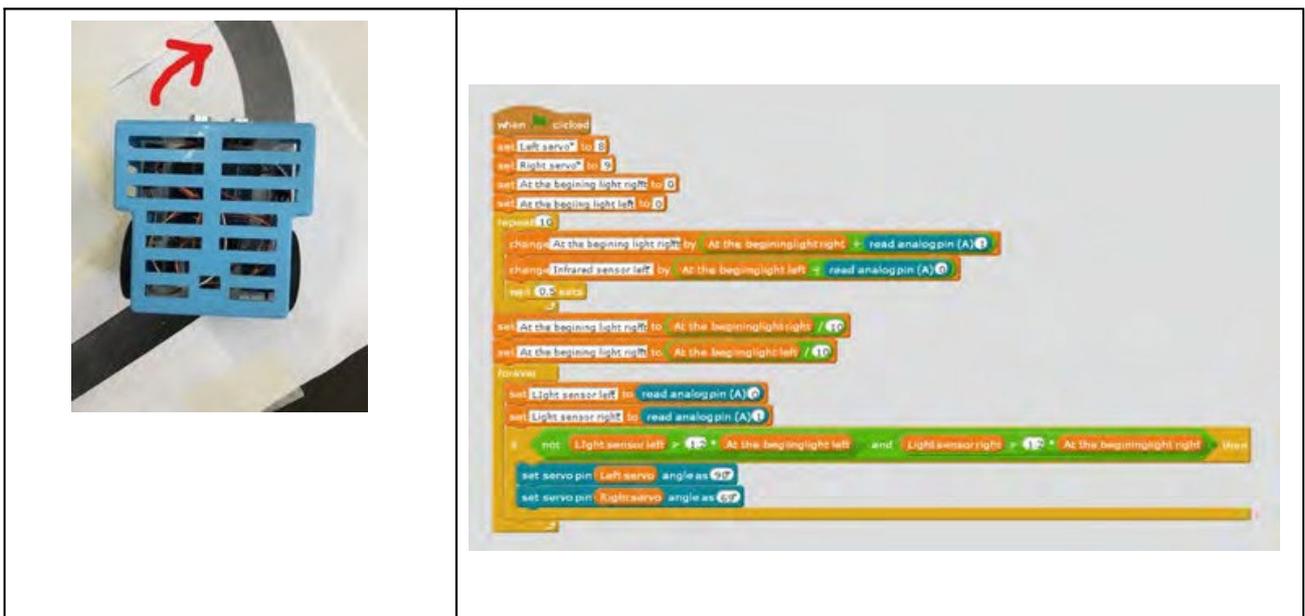
ACTIVIDAD 9



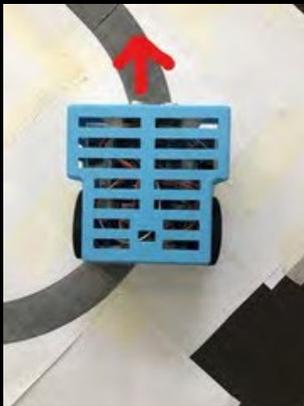
```
Programa de Arduino
fijar Left servo a 8
fijar Right servo a 9
fijar At the beginning light right a 0
fijar At the beginning light left a 0
repetir 10
  cambiar At the beginning light right por At the beginning light right + leer pin analógico (A) 1
  cambiar Infrared sensor left por At the beginning light left + leer pin analógico (A) 0
  esperar 0.5 segundos
fijar At the beginning light right a At the beginning light right / 10
fijar At the beginning light left a At the beginning light left / 10
```

- Entonces comparará el valor de la luz ambiental con la luz recibida por cada sensor, cuando la luz recibida sea mayor que la ambiental, esa rueda avanzará.
- Hay al menos tres situaciones posibles: Posteriormente comparará el valor de la luz ambiente con la luz recibida por cada sensor, cuando la luz recibida sea mayor que la luz ambiente, esa rueda caminará hacia delante.

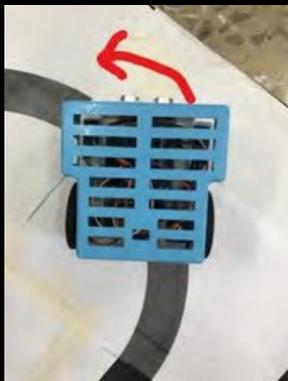
Son posibles al menos tres situaciones:



ACTIVIDAD 9



```
when clicked
  set Left servo to 0
  set Right servo to 0
  set At the begining light right to 0
  set At the begining light left to 0
  repeat 10
    change At the begining light right by At the begininglight right + read analog pin (A)
    change Infrared sensor left by At the begininglight left + read analog pin (A)
    wait 0.5 secs
  end
  set At the begining light right to At the begininglight right / 10
  set At the begining light left to At the begininglight left / 10
  forever
    set Light sensor left to read analog pin (A)
    set Light sensor right to read analog pin (A)
    if Light sensor left > 100 * At the begininglight left and not Light sensor right > 100 * At the begininglight right then
      set servo pin Left servo angle as 120
      set servo pin Right servo angle as 30
```



```
when clicked
  set Left servo to 0
  set Right servo to 0
  set At the begining light right to 0
  set At the begining light left to 0
  repeat 10
    change At the begining light right by At the begininglight right + read analog pin (A)
    change Infrared sensor left by At the begininglight left + read analog pin (A)
    wait 0.5 secs
  end
  set At the begining light right to At the begininglight right / 10
  set At the begining light left to At the begininglight left / 10
  forever
    set Light sensor left to read analog pin (A)
    set Light sensor right to read analog pin (A)
    if Light sensor left > 100 * At the begininglight left and Light sensor right > 100 * At the begininglight right then
      set servo pin Right servo angle as 120
      set servo pin Left servo angle as 30
```

Es aconsejable probar cada una de estas situaciones por separado, para saber si cada una de ellas se ha programado correctamente.

ACTIVIDAD 9

3. El programa final será

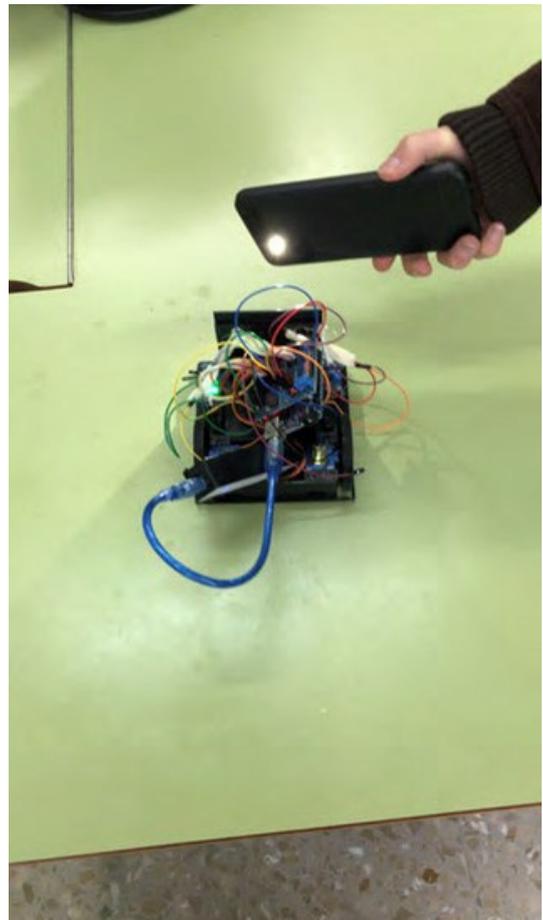
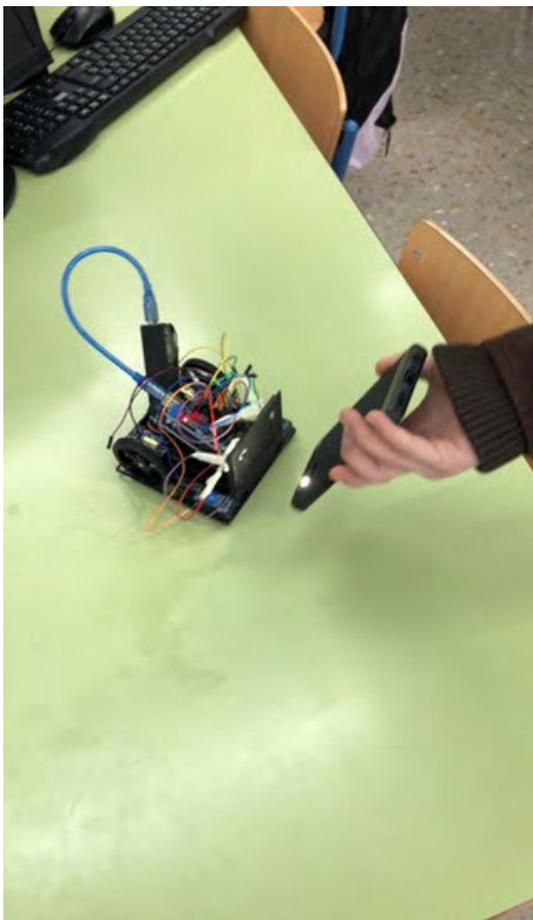
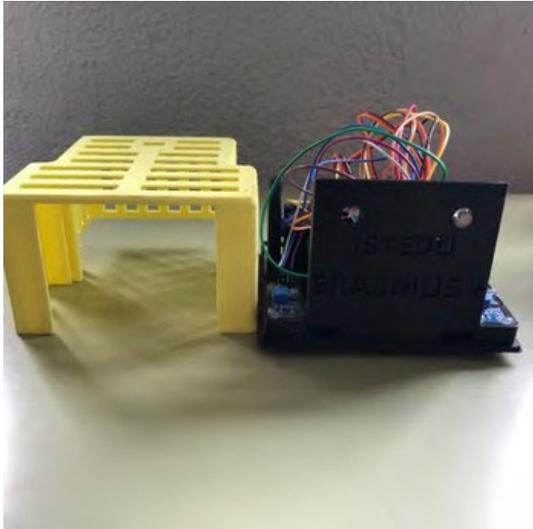
```
when clicked
set Left servo to 8
set Right servo to 9
set At the beginning light right to 0
set At the beginning light left to 0
repeat 10
  change At the beginning light right by At the beginning light right + read analog pin (A) 1
  change Infrared sensor left by At the beginning light left + read analog pin (A) 0
  wait 0.5 secs
set At the beginning light right to At the beginning light right / 10
set At the beginning light left to At the beginning light left / 10
forever
  set Light sensor left to read analog pin (A) 0
  set Light sensor right to read analog pin (A) 1
  if Light sensor left > 10 * At the beginning light left and Light sensor right > 10 * At the beginning light right then
    set servo pin Right servo angle as 125
    set servo pin Left servo angle as 75
  if Light sensor left > 10 * At the beginning light left and not Light sensor right > 10 * At the beginning light right then
    set servo pin Left servo angle as 125
    set servo pin Right servo angle as 90
  if not Light sensor left > 10 * At the beginning light left and Light sensor right > 10 * At the beginning light right then
    set servo pin Left servo angle as 90
    set servo pin Right servo angle as 65
```

Activar Windows

RECURSOS



ACTIVIDAD 9



EVALUACIÓN DEL ALUMNO

Se evaluará:

- Impresión de las diferentes partes del robot.
- Instalación de los componentes electrónicos.
- Funcionamiento correcto de todos los componentes electrónicos mediante la programación individual de cada uno de ellos.
- Programación de los diferentes movimientos del robot.
- El robot puede seguir una línea recta, una curva a la derecha o una curva a la izquierda.

BIBLIOGRAFÍA

<https://www.arduino.cc/>

ADAPTACIÓN A OTROS NIVELES

Es posible crear un circuito con obstáculos y hacer que el robot los evite. Por otro lado, la velocidad del robot puede variar en función de la luz recibida en comparación con la luz ambiente.

ACTIVIDAD 10

TÍTULO Quita nieves

RESUMEN

La rápida evolución de la tecnología ha alcanzado un nivel vertiginoso en los últimos años. Mientras que muchos objetos y tecnologías mencionados en novelas y películas de ciencia ficción en los años 70 y 80 se consideraban productos de la imaginación, ahora están en nuestras vidas. Los robots, que se consideraban sirvientes cuando se introdujo por primera vez el término robot, se utilizan hoy en muchos campos, desde la industria hasta el ejército. Aunque todavía no es muy común, la educación ha empezado a ser un ámbito en el que se utilizan robots. Los robots en la educación; Se utiliza como profesor, material didáctico, tutor de compañeros y profesor asistente. Se han realizado estudios sobre el uso de robots en muchas áreas de la educación.

Los estudios realizados demuestran que el uso de robots en la educación es eficaz y aumenta la motivación. Con el aumento de la investigación en este campo y la disminución de los costes de producción, será posible ver robots en muchos campos de la educación en un futuro próximo.

En este proyecto, diseñamos Mbot como un quitanieves. Observamos las máquinas quitanieves, de vital importancia en los días de invierno, desde los ojos de nuestros alumnos.

AUTOR/S

Sultantepe Prof. Dr. Cemil Taşçioğlu Ortaokulu

FECHA 26/01/2022

VERSIÓN 1

ACTIVIDAD 10

OBJETIVOS DIDÁCTICOS

Alumno principiante:

- Qué es mbot, cómo funciona
- Reconocer el programa Mblock
- Reconocer el software Tinkercad.

Saber lo que es una impresora 3D.

Escribir código, interfaz del programa, preferencias de idioma, conexión en el programa Mblock.

- Conocer la salida en formato .stl en la plataforma Tinkercad.
- Saber utilizar el programa de corte para impresora 3D.

CIENCIA

TECNOLOGÍA

MATEMÁTICAS

GEOGRAFÍA/HISTORIA

IDIOMAS

LITERATURA

MÚSICA

OTROS

NIVEL DE ESTUDIOS

Esta actividad está preparada para...

12 - 14 AÑOS

14 - 16 AÑOS

OTROS

HERRAMIENTAS NECESARIAS

Mbot(Bluetooth)

Mblock

Programa Mbot

Remote

Impresora 3d

Cardboard

Cinta adhesiva, pegamento, tijeras, algodón, silicona caliente

Maqueta de casa, maqueta de árboles

Ordenador (para modelar con Tinkercad)

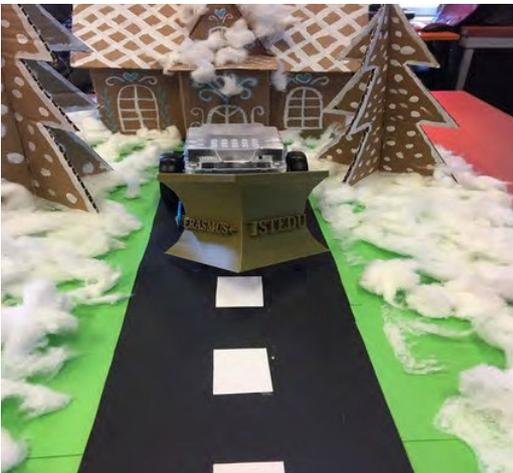
DESARROLLO DE LA ACTIVIDAD

A continuación se describe paso a paso la configuración del proyecto.

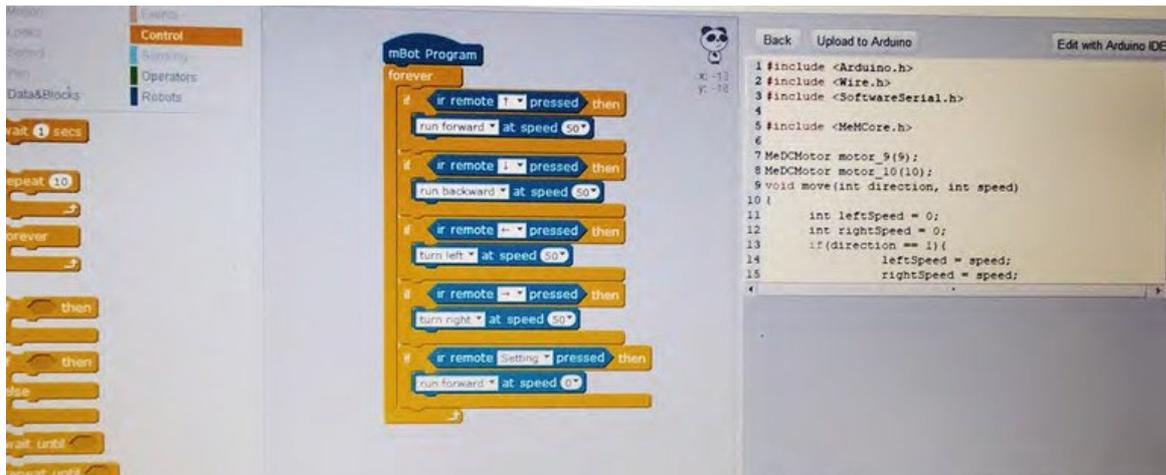
1. Se cortan los cartones en tamaños adecuados. Es preferible utilizar cartones de distintos colores para indicar que el suelo y la carretera están separados entre sí.
2. Las maquetas de casas y árboles se pegan sobre cartón.
3. Los algodones que representan la nevada se distribuyen en el suelo de cartón de tal forma que también cierra los caminos.
4. De acuerdo con las dimensiones de la parte delantera del Mbot, el aparato quitanieves se diseña en el software Tinkercad.
5. El diseño se imprime utilizando una impresora 3D, un programa de corte y filamento PLA. (Debe evitarse el contacto con las manos durante la impresión 3D. El filamento derretido será de 200 0C para el medio ambiente). La impresión impresa se monta en el Mbot.
6. En el programa Mblock, se escribe un programa para que Mbot pueda controlarse con un mando a distancia.
7. Mbot se le permite limpiar la carretera de la nieve en el cartón.

ACTIVIDAD 10

RECURSOS



ACTIVIDAD 10



EVALUACIÓN DEL ESTUDIANTE

Áreas en las que nuestros alumnos se evaluarán a sí mismos:

1. ¿Puedo utilizar el software Mblock para Mbot?
2. ¿Puedo modelar con Tinkercad?
3. ¿Puedo utilizar correctamente los materiales del proyecto y crear el terreno adecuado?

BIBLIOGRAFÍA

<https://ide.mblock.cc/>

<https://www.tinkercad.com/>

ADAPTACIÓN A OTROS NIVELES

Este proyecto puede considerarse de nivel principiante-intermedio para alumnos de quinto y sexto curso. De acuerdo con el tema, este proyecto;

Evade cuando Mbot ve un obstáculo utilizando el sensor de distancia. Se puede desarrollar para evitar que se salga de la carretera utilizando un sensor de seguimiento de línea, se puede hacer autónomo. Además, se puede acoplar un remolque a la parte trasera del quitanieves, conectar un servomotor a la pieza impresa en 3D y transferir la nieve recogida al remolque a intervalos regulares. De este modo también se eliminan las acumulaciones de nieve al borde de la carretera.