



PC Factory
Área de profesionales y especialistas
Equipo de computación y electrónica

Manual de usuario

Makeblock Software

mBlock, Makeblock App y mBot App



Índice

1	Introducción.....	2
2	mBlock	2
2.1	Requerimientos de sistema.....	2
2.2	Interfaz.....	2
2.3	Programación	3
2.4	Consideraciones.....	7
3	Aplicaciones para dispositivos portátiles.....	7
3.1	Makeblock App	7
3.1.1	Requerimientos de sistema	7
3.1.2	Interfaz	8
3.1.3	Programación	8
3.1.4	Consideraciones	9
3.2	mBot App	9
3.2.1	Requerimientos de sistema	10
3.2.2	Interfaz	10
3.2.3	Programación	11
4	Referencias.....	11

1. Introducción

Makeblock cuenta con una diversa gama de programas diseñados para interactuar con sus dispositivos, dentro de estos se encuentran aplicaciones (*Apps*) para dispositivos portátiles las cuales son descargables sin costo a través de alguna App Store. Además, un software para computadores, mBlock, en el cual se pueden programar en mayor detalle los diferentes robots. Todas los software de Makeblock vienen localizados en varios idiomas diferentes, los cuales incluyen español e inglés.

2. mBlock

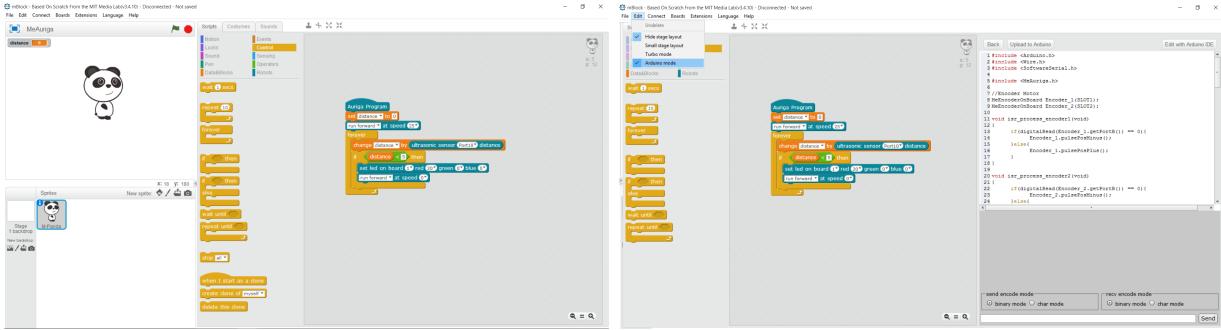
mBlock es un software desarrollado en base a Scratch [1] y Arduino [2] para la programación de diversos robots producidos por Makeblock, este se puede descargar de forma gratuita a través de la pagina de <http://learn.makeblock.com/en/software/>. Un detallado manual con todas las instrucciones de instalación, mayor detalle de algunos bloques y un ejemplo de uso de este software puede ser encontrado en <http://download.makeblock.com/mblock/docs/getting-started-with-mblock.pdf>

2.1. Requerimientos de sistema

- Sistema operativo: MAC OS x64, Windows XP x64 y superiores, Chrome OS, Linux x86.
- Espacio en memoria: 463MB disponibles.

2.2. Interfaz

La interfaz inicia en un modo donde se presentan dos secciones principales, una en donde se puede trabajar con sprites de imágenes y otra que es la base de la programación por bloques, basada de Scratch. Además de esto, existe la opción de abrir una ventana en donde se muestra el código Arduino que hay por detrás de cada bloque de Scratch, con la posibilidad de editar dicho código de forma directo en el IDE (entorno de desarrollo integrado) de Arduino.

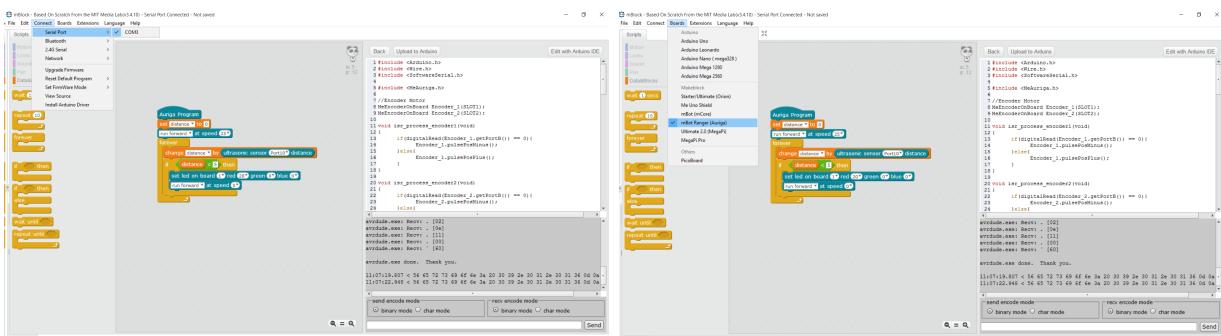


(a) Pantalla de inicio

(b) Código Arduino

Figura 1: mBlock Software

Al momento de utilizar alguno de los dispositivos de Makeblock, se puede seleccionar alguna de las opciones mostradas en la Figura 2 para especificar el tipo de conexión y robot utilizado.



(a) Selección conexión. Disponibles: Serial Port, (b) Selección dispositivo. Disponibles: Starter/Ultimate, Me Uno Shield, mCore, Auriga, Ultimate 2.0 y MegaPi Pro

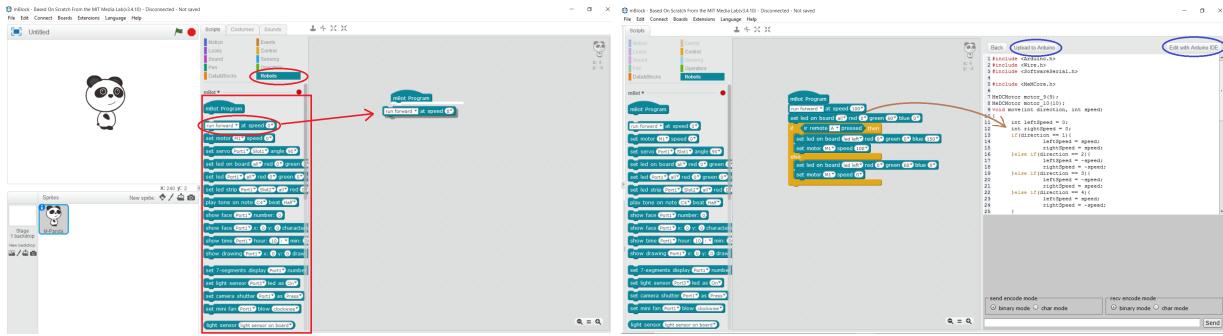
Figura 2: Ejemplo de conexión a través del puerto serial COM3 con un mRanger (Auriga)

2.3. Programación

mBlock soporta un estilo de programación de forma nativa y este corresponde al basado en Scratch, además cuenta con la opción de extender la edición de programas a el escribir directamente el código de estos, esto a través de una portabilidad mBlock-IDE Arduino.

La primera modalidad de programación a corresponde a la basada en bloques. Para el uso de esta se tiene una variedad de bloques dentro de la sección **Script** del programa, como se puede ver en la Figura 3 (a), estos tienen diversas formas y colores los que se encuentran asociados a su función. El procedimiento para desarrollar un programa para los robots consiste en el seleccionar bloques de la lista, después arrastrarlos con el cursor hasta el

espacio vacío adyacente y luego unirlos entre ellos usando su naturaleza de rompecabezas. De esta manera, se pueden ir agregando diferentes funciones (bloques) a lo que será el programa a probar en el robot. Para que estas sean ejecutables deben estar unidas a un bloque central llamado **Arduino Program** o **<inserte robot> Program** (en el caso de la Figura 3 (b) se utiliza un mBot, i.e. se utiliza **mBot Program**). La selección de este bloque central esta asociada a que tipo de bloques se quiere ocupar, si aquellos en terminología de Makeblock o su equivalente en Arduino.



(a) Sección **Script**, con los bloques para programar. (b) Código Arduino correspondiente a los bloques Animación del proceso disponible en [https://goo.g](https://goo.gl/HNQffC)l/HNQffC del programa

Figura 3: Dinámica de programación.

Este método de programación basado en bloques esta diseñado para ser intuitivo y amigable para un usuario con poca o nula experiencia en el mundo de la programación. En cuanto a la funcionalidad, los bloques presentes en el programa abarcan a cabalidad las funciones básicas asociadas a los sensores y actuadores presentes en cada robot de Makeblock. Con esto se pueden desarrollar tantos algoritmos como se deseé.

La segunda modalidad, consiste en programar en el IDE de Arduino haciendo uso de la portabilidad presentada en la Figura 3 (b). Esto implica el escribir código compatible con los microcontroladores presentes en los robots, dicho código debe estar escrito ya sea en el lenguaje propio de Arduino o en lenguaje C. Para poder programar los dispositivos de Makeblock existen una serie de librerías con funciones creadas para utilizar los diferentes sensores y actuadores de los robots. Como se puede ver en la Figura 4, existe una amplia variedad de librerías para Arduino para hacer uso de las funcionalidades de los robots. La documentación de como utilizar estas librerías se encuentra distribuida a lo largo de las páginas de Makeblock, en estas se encuentra un compilado de las librerías el cual incluye los constructos y definiciones de las estructuras de funciones en <http://learn.makeblock.com/Makeblock-library-for-Arduino/index.html>. De forma complementaria, en <http://learn.makeblock.com/en/> se encuentra información acerca de los esquemáticos, modo de conexión y ejemplos en Arduino de algunos módulos electrónicos y tarjetas de Makeblock (ver ejemplo en la Figura 5).

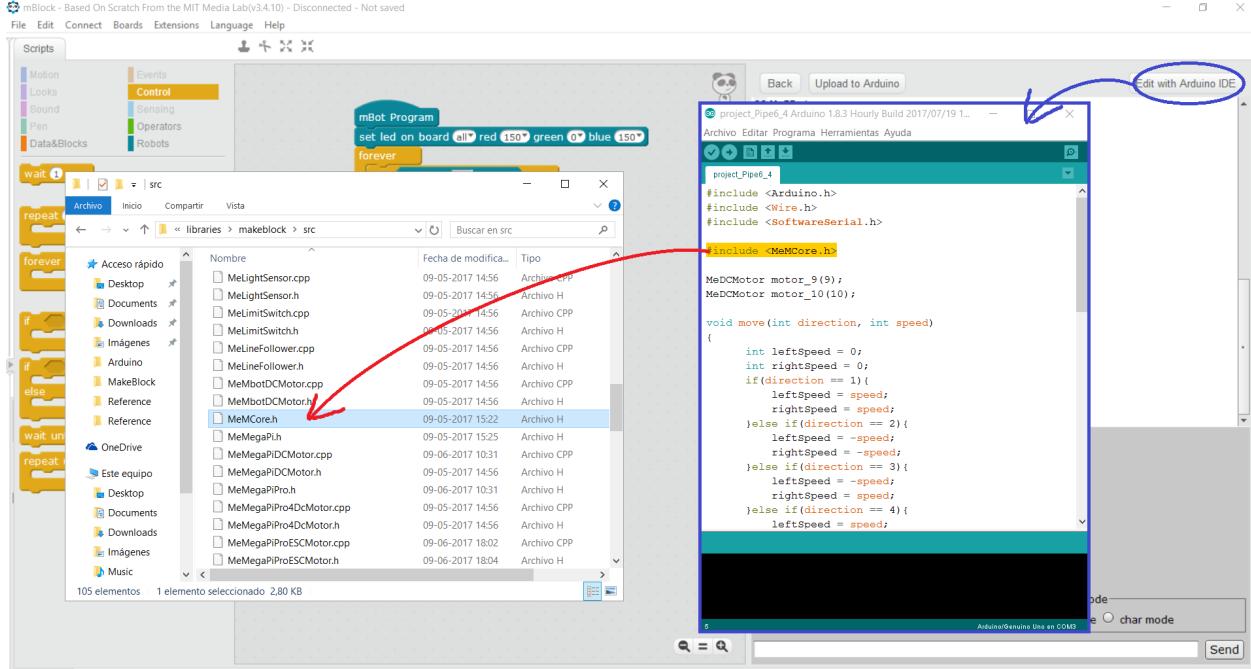


Figura 4: Librerías de Makeblock

Para complementar la experiencia del usuario y la función educativa de este programa, existen libros (3 en inglés y 1 en español) [6] que explican el razonamiento y proceso detrás de la creación de un algoritmo con el software. Estos están diseñados para convertirse en los libros guías de cursos de nivel escolar y están disponibles en <http://www.mblock.cc/edu/>, con la excepción del 4^{to} que proviene de la sección de España de Makeblock (<https://www.makeblock.es/soporte/robot-mbot/>). Por otro lado, Makeblock cuenta con una plataforma de soporte activo a través de su foro de consultas, en <http://forum.makeblock.com/>, y por medio de la sección de preguntas en cada uno de sus productos en la tienda.



Figura 6: Libros gratuitos de Makeblock

Me Potentiometer



Overview

Featuring an adjustable range with the highest resistance of 10 KΩ, the Me Potentiometer module is a resistor with three terminals, and its resistance can be adjusted by rotating the knob. The module can be used to adjust rotational speed of motor, and brightness of LED lamp. Its black ID means that it has an analog signal port and needs to be connected to the port with black ID on Makeblock Orion.

Technical specifications

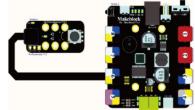
- Operating voltage: 5V DC
- Maximum current: 30 mA
- Rated power: 0.1W
- Insulation resistance: 100MΩ
- Total resistance: 10 KΩ
- Signal type: analog signal (0~980)
- Module size: 51 x 24 x 22 mm (L x W x H)

(a) Presentación del módulo y especificaciones técnicas

Wiring mode

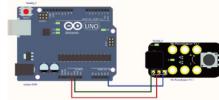
Connecting with RJ25

Since the port of Me Potentiometer has black ID, you need to connect the port with black ID on Makeblock Orion when using RJ25 port. Taking Makeblock Orion as example, you can connect to ports No. 6, 7, and 8 as follows:



Connecting with Dupont wire

When the Dupont wire is used to connect the module to the Arduino UNO Baseboard, its AOP pin should be connected to analog pin as follows:



(c) Modo de conexión a tarjeta

Functional characteristics

- A blue LED is provided on the module, and its brightness change reflects the change of analog output value
- White area of module is the reference area to contact metal beams
- Support Arduino IDE programming, and provide a runtime library to simplify the programming
- Support mBlock GUI programming, and applicable to users of all ages
- Adapt RJ25 port for easy connection
- Provide pins to support most Arduino Baseboards

Pin definition

The port of Me Potentiometer has three pins, and their functions are as follows:

No.	Pin	Function
1	GND	Grounding
2	VCC	Power supply
3	AO	Analog output of potentiometer (voltage range: 0~4.8V)

(b) Función y mapeo de pines

Guide to programming

Arduino programming

If you use Arduino to write a program, the library Makeblock-Library-master should be invoked to control and read the output value of Me Potentiometer.

```
#include "MePoti.h"
#include <Wire.h>
#include <SoftwareSerial.h>

MePoti mePoti(0);

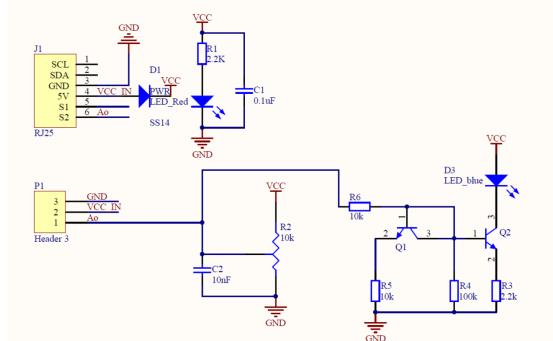
void setup()
{
    Serial.begin(9600);
}

void loop()
{
    Serial.print("Potentiometer=");
    Serial.println(mePoti.read());
    delay(100);
}
```

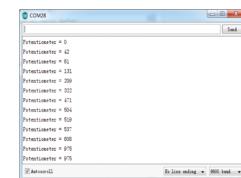
Function name	Function
MePoti(uchar port)	Select a port
int16_t read()	Read the output value of potentiometer (0~980)

(d) Ejemplo de código y función asociada

Schematic



The function of the code segment is: to read the parameter of Me Potentiometer and output the result to the serial monitor in Arduino IDE in the cycle of 100 ms. Upload the code segment to the Makeblock Orion and click the Arduino serial monitor, and then you will see the running result as follows:



The value range of potentiometer is 0~980. It decreases when you rotate the knob counterclockwise and increases when you rotate it clockwise.
mBlock programming
Me Potentiometer supports the mBlock programming environment and its instructions are introduced as follows:

Programming description	Description
potentiometer Port7	Parameters: Select a port Equation: Return the position parameter of potentiometer (0~980)

(e) Resultado y bloque mBlock equivalente

(f) Esquemático del módulo

Figura 5: Documentación módulo electrónico MePotentiometer

2.4. Consideraciones

Debido al modo de funcionamiento del software, es posible que el usuario tenga complicaciones con algunos aspectos. Aquellos que amenazan el correcto funcionamiento son los relacionados a permisos y compatibilidad de programas, para evitar esto se han de tener las siguientes consideraciones:

- El software mBlock DEBE ser ejecutado en **Modo Administrador**, para que no presente problemas la portabilidad con Arduino.
- El PC debe tener una versión de Java compatible con la de 32bit. A pesar de esto, puede que sea necesario reemplazar la versión de Arduino embebida en el programa mBlock, para esto basta con realizar una copia de la carpeta **portable** ubicada en **C:\Program Files (x86)\mBlock\Arduino** y pegarla en la nueva carpeta de Arduino que se instale. Otra opción corresponde a agregar el paquete de liberías makeblock a la carpeta de librerías de Arduino, de manera que termine existiendo la ruta **Arduino\libraries\makeblock\src**.

3. Aplicaciones para dispositivos portátiles

Makeblock cuenta con dos aplicaciones para dispositivos portátiles, estas son **Makeblock** y **mBot**. La primera es la aplicación principal de la empresa, la cual sirve para conectarse a través de Bluetooth con los diferentes robots, por esto requieren estar a una distancia prudente tal que haya conexión. Con esta se pueden enviar instrucciones y programar algunos algoritmos para que los robot los ejecuten. La segunda corresponde a una App desarrollada con el propósito de enseñar *cómo* programar a los niños, esta es utilizable solo con el modelo inicial de Makeblock **mBot**.

3.1. Makeblock App

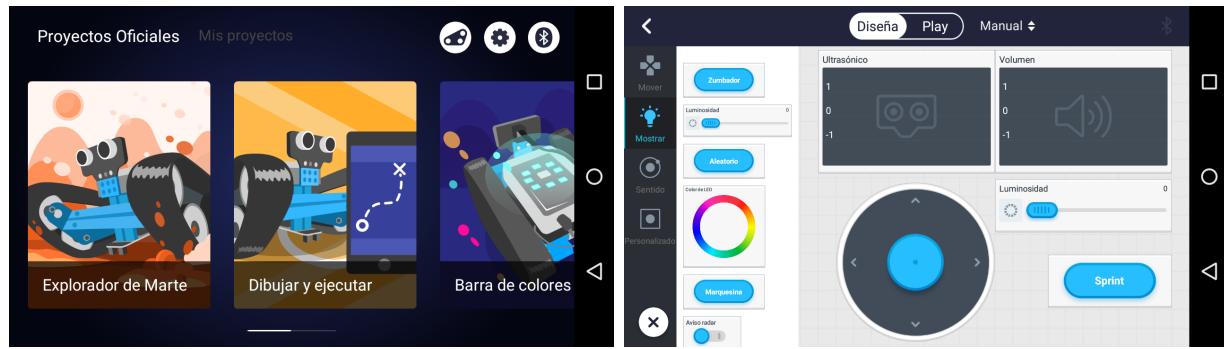
Esta aplicación tiene como objetivo principal el servir como nexo entre el usuario y los diferentes robots de la empresa.

3.1.1. Requerimientos de sistema

- Sistema operativo: **iPhone4S/iPad 3 o superior, iOS 9.0 o superior, Android 4.3 o superior.**
- Espacio en memoria: 122MB disponibles.

3.1.2. Interfaz

La aplicación parte con una pantalla de sincronización con el robot, con la que posteriormente se abre un menú de aplicaciones de acuerdo al robot detectado. Una vez dentro del menú, se presenta una gama de aplicaciones pre-programadas para hacer un uso demostrativo de las capacidades de cada modelo (ver Figura 7 (a)). En la Figura 7 (b) se muestra la interfaz correspondiente al menú **Crear** de la App, en este se cuenta con módulos que están programados para hacer uso de los sensores y actuadores del robot en uso.



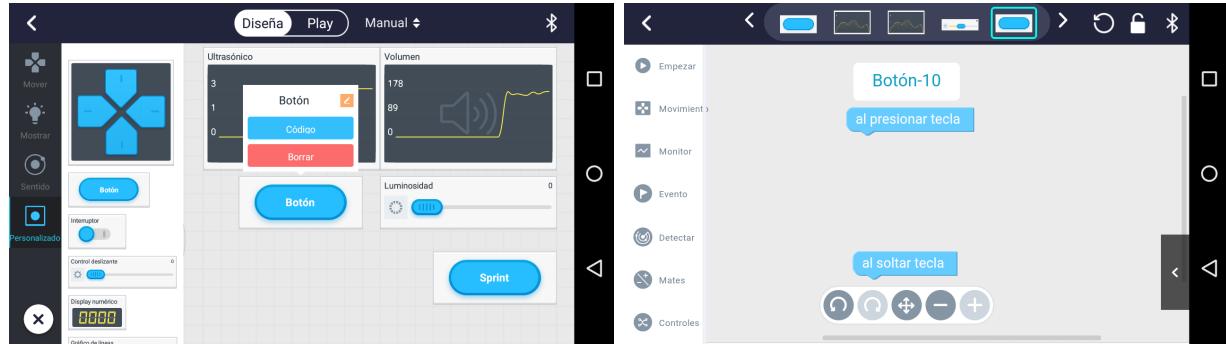
(a) Menú para modelos dentro de la aplicación

(b) Menú **Crear**, aplicación diseñable

Figura 7: Interfaz Makeblock App

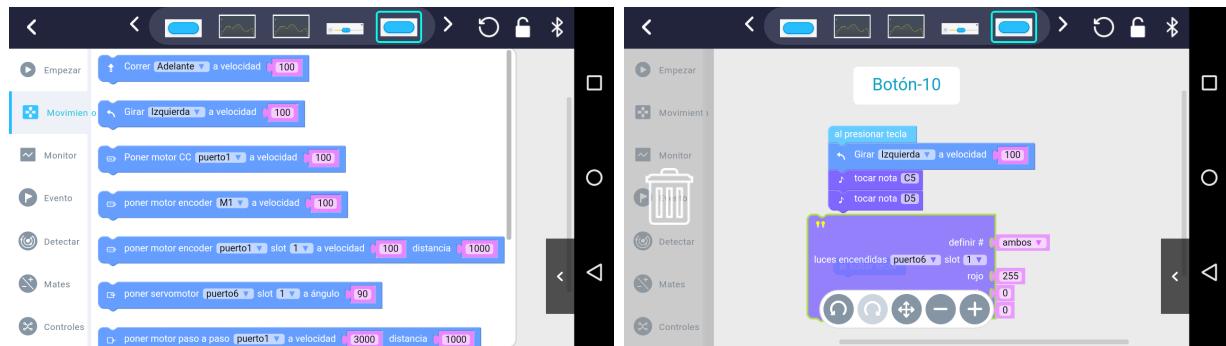
3.1.3. Programación

Al ser seleccionados, la mayoría de los módulos dentro del menú **Crear** tienen la opción para abrir su **Código** (Figura 8 (a)). Al abrir el código se entra en una interfaz como la mostrada en la Figura 8 (b), en donde se tiene un menú a la izquierda en el cual se encuentran bloques de mBlock para la programación del modulo. Un ejemplo del menú y el uso de estos bloques se puede ver en (c) y (d) de la Figura 7. De forma análoga a lo que es la programación en el software para PC, todos los bloques seleccionados deben estar unidos entre sí y bajo alguna directriz inicial que los active, como lo es el bloque **al presionar tecla**.



(a) Opciones de un módulo botón

(b) Interfaz Código



(c) Bloques del menú Movimiento para utilizar

(d) Ejemplo de programa en bloques

Figura 8: Interfaces y menús para programar en la App

3.1.4. Consideraciones

- El funcionamiento de esta aplicación está sujeto a la señal Bluetooth utilizada, es decir, tiene una distancia máxima de operación dependiente del modelo de robot y celular utilizado. En la actualidad, la mayoría de los productos Makeblock cuentan con un módulo Bluetooth de 100m de alcance al aire libre.
- En condiciones normales la App no presenta niveles de lag apreciable que puedan afectar la experiencia del usuario.
- A veces las instrucciones de giro se interpretan de forma intercambiada, es decir, se confunde (\rightarrow) por (\leftarrow) y viceversa.

3.2. mBot App

La aplicación mBot tiene como objetivo el enseñar a niños de forma didáctica el *cómo* programar, particularmente el robot mBot.

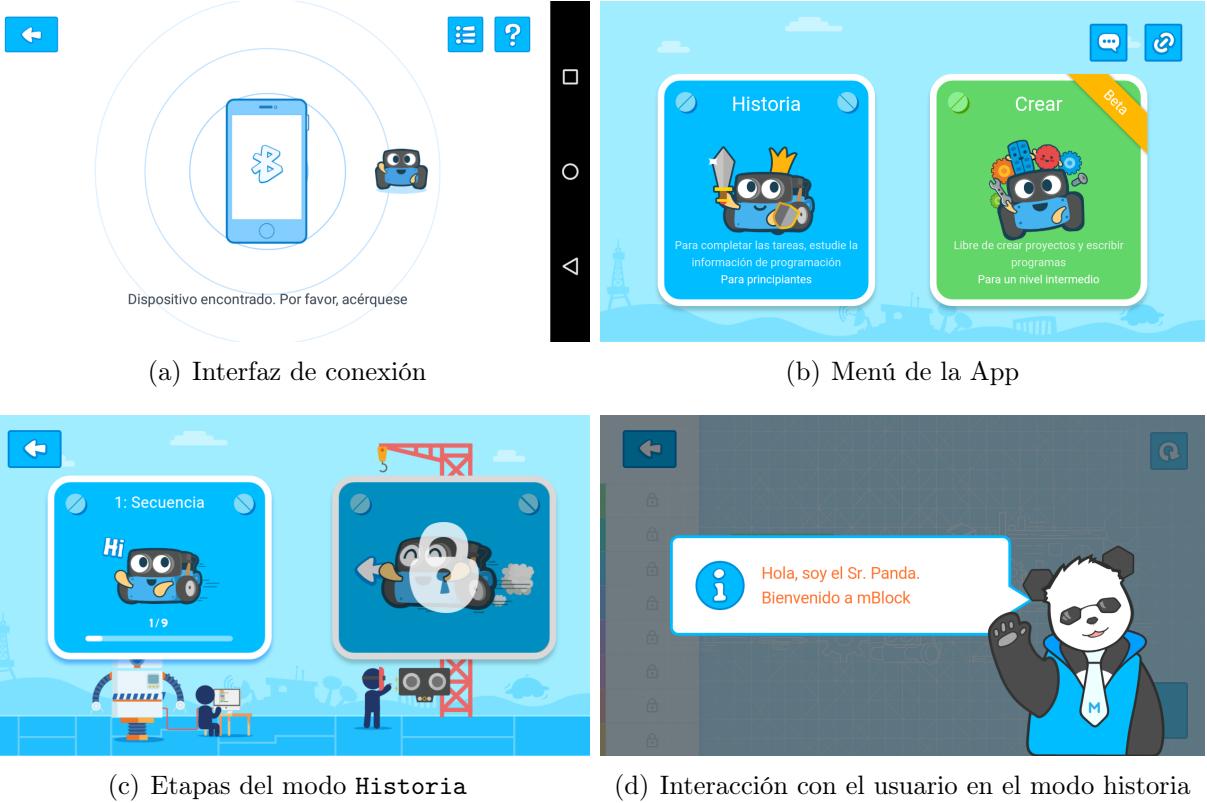


Figura 9: Interfaz mBot App

3.2.1. Requerimientos de sistema

- Sistema operativo: iPhone4S/iPad 3 o superior, iOS 9.0 o superior, Android 4.3 o superior.
- Espacio en memoria: 110MB disponibles.

3.2.2. Interfaz

La aplicación inicia con una pantalla de sincronización con el robot (Figura 9 (a)), con la que posteriormente se abre un menú con los modos principales de la App (Figura 9(b)). El modo **Crear** consiste en una interfaz para programar en bloques, equivalente a la de la App principal de Makeblock, mostrada en la Figura 8. Por otro lado, el modo **Historia** presenta una serie de etapas para que el usuario vaya superando, estas se van desbloqueando a medida que el usuario va completando las misiones previas (Figura 9(c)). Dentro de estas etapas existe una interacción activa con el usuario para guiarlo a través de la aplicación, como se puede ver en la (Figura 9(d)).

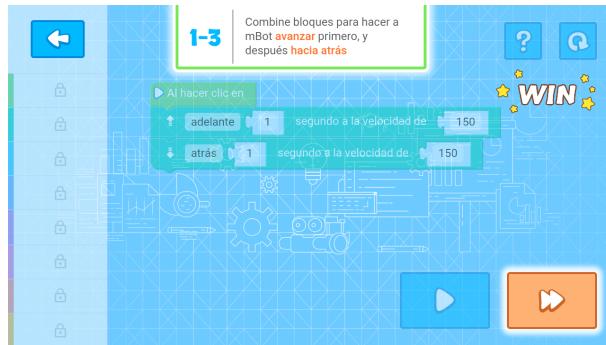


Figura 10: Programación en modo historia

3.2.3. Programación

El formato de corresponde al mismo de los otros software de Makeblock y corresponde a la programación por bloques, es accesible de forma directa a través del modo **Crear**. El *cómo* utilizar los bloques para la programación corresponde al foco principal del modo **Historia**, y al de la App en general. Es en este modo en que se enseña al usuario, a través de etapas como la mostrada en la Figura 10, los fundamentos de la programación y el uso de bloques para construirla.

Esta modalidad ha sido desarrollada con un estilo visual que resulte lo más llamativo y amigable para niños de edad escolar.

4. Referencias

Referencias

- [1] Scratch, software con propósitos educativos. Proyecto del grupo Lifelong Kindergarten del MIT Media Lab. Disponible en:
<https://scratch.mit.edu/>
- [2] Arduino, plataforma electrónica open-source diseñada para el desarrollo de diversos proyectos interactivos. Disponible en:
<https://www.arduino.cc/>