UDEC
UNIVERSIDAD DE CUNDINAMARCA

TALLER 02

PROGRAMACIÓN DEL ROBOT "INSPIRABOT" CON MBLOCK

El presente Taller 02 forma parte de la mediación pedagógica en robótica educativa, cuyo propósito es integrar el aprendizaje STEAM mediante la construcción, calibración y programación de un robot físico. En la fase anterior, los participantes lograron ensamblar y configurar InspiraBot, asegurando su correcto funcionamiento mecánico y electrónico. Ahora, en esta nueva etapa, explorarán los fundamentos de la programación en mBlock, una plataforma basada en bloques que facilita la lógica computacional y la interacción con los sensores y actuadores del robot. Este taller está diseñado para promover el aprendizaje significativo a través de la experimentación, la resolución de problemas y la aplicación de metodologías activas. A lo largo de la sesión, los participantes desarrollarán habilidades en pensamiento computacional, estructuración de código, depuración de errores y lógica algorítmica, lo que les permitirá programar el movimiento y la toma de decisiones del robot en distintos escenarios.

Durante la sesión, los participantes trabajarán en equipos para diseñar y probar secuencias de comandos en mBlock, explorando cómo los sensores permiten al robot percibir su entorno y responder de manera autónoma. Se abordarán conceptos clave como estructuras de control (condicionales y bucles), interacción con sensores de línea y ultrasonido, y ejecución de movimientos programados. Al finalizar, los participantes habrán programado InspiraBot para ejecutar tareas específicas y enfrentar un desafío final que pondrá a prueba su lógica de programación y su capacidad de optimización del código.

Duración: 5 horas (2 presenciales- 3 virtuales)
Edad recomendada: 13 - 20 años

Participantes: 3 integrantes por equipo

Participante 1: _____
Participante 2: _____
Participante 3: _____

Habilidades para desarrollar:

- Resolución de problemas
- Trabajo en equipo
- Pensamiento lógico

Módulos:

1. Configuración de mBlock para programar el robot
2. Encender y apagar LED del robot
3. Leer sensores y tomar decisiones
4. Control de motores y movimientos
5. Programación de comportamientos autónomos

Elaborado por: Javier Hernando Ruiz Farfán

Introducción: Planteamiento del Problema (15 min)

Situación Inicial:

" Imagina que eres parte de un equipo de ingenieros que trabaja en la automatización de un robot para explorar una zona de difícil acceso. El reto es programar el robot para que realice tareas autónomas como detectar obstáculos, seguir una ruta y responder a estímulos del entorno. Para ello, primero debemos aprender a programarlo desde cero con mBlock."

Preguntas de reflexión inicial:

a) ¿Cómo puede un robot "tomar decisiones"?
b) ¿Qué sensores y actuadores podríamos utilizar para que un robot interactúe con su entorno?
c) ¿Qué pasos crees que se deben seguir para programar un robot desde cero?

MODULOS

1. CONFIGURACIÓN DE MBLOCK PARA PROGRAMAR EL ROBOT (15 MIN)

Objetivo: Instalar y conectar correctamente el robot a mBlock.

Materiales:

- Piezas estructurales del robot (material reciclable, PLA, metal)
- Motores
- Sensores (ultrasonidos, sensores de línea, bumpers, sonido)
- Tornillos y herramientas necesarias
- Baterías 18650
- PC con mblock instalado

Actividades:

a) Instalación y configuración de **mBlock** en el PC. [Ver cartilla]
b) Conexión del robot y verificación de comunicación con el software. [Ver cartilla]
c) Prueba de conexión con un código de diagnóstico. [Ver cartilla]

Desafío

Elaborado por: Javier Hernando Ruiz Farfán

Cada equipo debe instalar y configurar mBlock, conectar el robot y ejecutar un código de prueba para verificar la comunicación entre el software y el hardware. Para superar el desafío, deben asegurarse de que el robot responda correctamente a los comandos básicos sin errores de conexión.

Puntos clave a observar (Facilitador):

- Verificar que los equipos completen la instalación de mBlock sin errores.
- Confirmar que los equipos logren enviar un comando básico al robot.
  Evaluar cómo los equipos identifican y corrigen errores de conexión.

_____
_____
_____
_____

Preguntas de reflexión:

¿Por qué es importante verificar la conexión antes de programar funciones avanzadas?

Diario de campo (Participante):

_____
_____
_____

2. ENCENDER Y APAGAR LED DEL ROBOT (20 MIN)

Objetivo: Comprender la lógica básica de programación en bloques.

Actividades:
Creación de un programa básico para encender y apagar un LED en el robot. Modificación del código para cambiar el tiempo de encendido y apagado.

Desafío:
Los equipos deben escribir un código que haga que el LED parpadee en intervalos personalizados, demostrando que comprenden la manipulación del tiempo en la programación.

Puntos clave a observar (Facilitador):

- Identificación y uso correcto del bloque de encendido/apagado.
- Modificación de tiempos de parpadeo según indicaciones.
- Uso de lógica secuencial en la estructura del programa.

_____
_____
_____

Preguntas de reflexión:

¿Cómo podríamos usar esto para señales visuales en un robot autónomo?

Diario de campo (Participante):

_____
_____
_____

3. LEER SENSORES Y TOMAR DECISIONES (30 MIN)

Objetivo: Programar el robot para procesar datos de sensores y reaccionar según condiciones predefinidas.

Actividades:

- Programación de lectura de sensores (ejemplo: ultrasonidos o sensores de línea).
- Implementación de una decisión basada en la lectura del sensor (Ej: encender un LED si hay un obstáculo).
- Prueba de diferentes escenarios y ajustes en los umbrales del sensor.

[Ver cartilla]

Desafío:
Los equipos deben programar el robot para detectar un obstáculo y tomar una decisión automática (Ej: detenerse o cambiar de dirección).

Puntos clave a observar (Facilitador):

- Verificar que los sensores de línea detecten correctamente el suelo y los valores sean ajustados según el entorno.
- Asegurar que los motores giren correctamente y el robot se desplace en línea recta sin desviaciones.
- Comprobar que los sensores de ultrasonido detecten obstáculos y generen la reacción esperada en el robot.

_____
_____
_____

Preguntas de reflexión:

a) ¿Cómo mejoraría la precisión del robot si combinamos más de un sensor?

Diario de campo (Participante):

_____
_____
_____

4. CONTROL DE MOTORES Y MOVIMIENTOS (30 MIN)

Objetivo: Programar el desplazamiento del robot controlando sus motores.

Actividades:

- Programación para mover el robot hacia adelante, atrás, girar a la izquierda y derecha.
- Introducción a la modificación de la velocidad de los motores.
- Prueba del movimiento en una pista de prueba.

Desafío:

Cada equipo debe programar el robot para moverse en un patrón cuadrado utilizando comandos de giro y desplazamiento.

Puntos clave a observar (Facilitador):

- Configuración de velocidad adecuada.
- Ajustes en la duración de los movimientos.
- Corrección de errores en la trayectoria del robot.

_____
_____
_____

Preguntas de reflexión:

¿Cómo podemos hacer que el robot siga una ruta sin intervención humana?

Diario de campo (Participante):

_____
_____
_____

5. PROGRAMACIÓN DE COMPORTAMIENTOS AUTÓNOMOS (25 MIN)

Objetivo: Integrar sensores y lógica condicional para que el robot tome decisiones sin intervención humana.

Actividades:

- Creación de una **secuencia de comandos** para que el robot se mueva de forma autónoma.
- Ajuste de parámetros para que el robot reaccione ante diferentes estímulos.
- Prueba y optimización del código en distintos escenarios.

Desafío:

Cada equipo debe programar el robot para seguir una ruta y evitar obstáculos de forma autónoma.

Puntos clave a observar (Facilitador):

- Integración de sensores con decisiones programadas.
- Estructuración lógica de los comandos en mBlock.
- Evaluación de la fluidez del comportamiento del robot.

_____
_____
_____

Preguntas de reflexión:

¿Cómo podríamos optimizar el código para mejorar el comportamiento del robot?

Diario de campo (Participante):

_____
_____
_____

6. RETO FINAL

Escenario:

Su misión es programar **InspiraBot** para que navegue de manera autónoma por un circuito determinado, detectando y esquivando obstáculos sin intervención humana.

Para ello, deberán integrar sensores de línea y ultrasonido en su código, permitiendo que el robot tome decisiones en tiempo real según las condiciones del entorno. La eficiencia de su programación determinará si el robot logra completar la ruta sin errores.

Desafío:

Cada equipo deberá programar el robot para:

- Seguir una ruta establecida sin intervención manual.
- Detectar obstáculos y reaccionar correctamente (detenerse o esquivar).
- Optimizar el código para evitar errores y mejorar el tiempo de ejecución.

Reglas del reto:

- Tiempo límite: 3 intentos para completar el recorrido correctamente.
- Uso de sensores: Debe utilizar sensores de línea para seguir la ruta y sensores de ultrasonido para detectar obstáculos.
- Código estructurado: Se evaluará que el código tenga una lógica clara y bien organizada.
- Fluidez del recorrido: El robot no debe detenerse innecesariamente ni salir del camino.
- Optimización: El equipo puede mejorar su código tras cada intento, ajustando parámetros según la observación de fallos.

Trabajo Independiente en Moodle (2 horas)

- **Actividad 1:** Reflexión escrita sobre lo aprendido en el taller.
- **Actividad 2:** Simulación en mBlock de una nueva tarea autónoma para el robot (ejemplo: esquivar varios obstáculos en secuencia).

- **Foro de discusión:** Compartir los desafíos enfrentados en la programación y cómo fueron solucionados.

Diario de campo (Participante):

_____

_____

_____

CIERRE Y EVALUACIÓN (10 MIN)- DOCENTE

Actividad: Reflexión y socialización de experiencias en equipo.

Cada equipo compartirá sus aprendizajes y responderá a las siguientes preguntas de análisis:

- Estrategias utilizadas en la programación: ¿Cómo estructuraron el código para que el robot siguiera la ruta y reaccionara ante obstáculos?
- Aplicación del pensamiento lógico: ¿Cómo identificaron y resolvieron los errores en la programación?
- Transferencia de conocimientos: ¿Cómo pueden aplicar este método en otras áreas de la tecnología, como domótica, automatización o inteligencia artificial?

Conclusión:
Este taller permitió a los participantes desarrollar su pensamiento computacional y lógico a través de la programación de InspiraBot en mBlock. Además, reforzó su capacidad para estructurar soluciones de manera secuencial, depurar errores y optimizar código, habilidades fundamentales en entornos STEAM.

AUTOEVALUACIÓN (PARTICIPANTE)

Cada participante evaluará su desempeño en una escala del 1 al 5, donde:

1 Nunca | 2 Pocas veces | 3 Algunas veces | 4 Casi siempre | 5 Siempre

Criterio	**Participante 1**	**Participante 2**	**Participante 3**
Participé activamente en todas las actividades.	☐	☐	☐
Cumplí con mi rol dentro del equipo.	☐	☐	☐
Ayudé a resolver problemas en la programación del robot.	☐	☐	☐
Trabajé en equipo y respeté las ideas de los demás.	☐	☐	☐

Reflexión rápida:

¿Qué mejorarías en un próximo taller?

Elaborado por: Javier Hernando Ruiz Farfán

UDEC
UNIVERSIDAD DE CUNDINAMARCA

_____
_____
_____
_____
_____

RÚBRICA DE EVALUACIÓN - TALLER 02

Criterio de Evaluación	**1 - Bajo**	**2 - Regular**	**3 - Aceptable**	**4 - Bueno**	**5 - Excelente**
Resolución de Problemas (Identifica fallos, propone soluciones y aplica correcciones en el ensamblaje, calibración o pruebas del robot).	No identificó ni intentó solucionar fallas.	Identificó fallas pero no logró solucionarlas de manera efectiva.	Identificó y corrigió fallas con ayuda del equipo o facilitador.	Identificó problemas y aplicó soluciones con pocas correcciones.	Identificó problemas y solucionó de manera autónoma y efectiva.
Trabajo en Equipo (Colabora con su equipo, respeta los roles y se comunica asertivamente).	No colaboró, conflictos con compañeros o no participó.	Participó de forma mínima o con dificultades de comunicación.	Participó con el equipo, aunque con algunas fallas en la comunicación.	Se integró activamente, colaboró y respetó los roles del equipo.	Demostró liderazgo, promovió la colaboración y facilitó el trabajo del equipo.
Pensamiento Lógico (Sigue pasos estructurados en el ensamblaje, configuración y calibración del robot).	No siguió una secuencia lógica en el ensamblaje o calibración.	Intentó seguir pasos pero con errores graves en la secuencia.	Siguió los pasos correctamente con algunas dificultades.	Aplicó los pasos de manera ordenada y eficiente.	Demostró una comprensión clara y optimizó la secuencia con lógica estructurada.