

## Robótica en FabLab: Introducción a la programación para estudiantes de Ingeniería Mecánica

Verónica Dangelo <sup>1</sup>, Guillermo Luján Rodríguez <sup>1,2</sup>, María Florencia Sklate <sup>2</sup> y Cesar Ignacio Pairetti <sup>2</sup>

<sup>1</sup> Instituto Rosario de Investigaciones en Ciencias de la Educación (IRICE: CONICET-UNR), Rosario, Argentina.

<sup>2</sup> Facultad de Ciencias Exactas, Ingeniería y Agrimensura (FCEIA: UNR), Rosario, Argentina.

verodangelo71@gmail.com

**Resumen.** En este trabajo se analiza una experiencia piloto de la actividad “Robótica” en el FabLab UNR, un laboratorio *spacemaker* creado en 2020 por la Facultad de Ciencias Exactas, Ingeniería y Agrimensura de la Universidad Nacional de Rosario, en el marco de las políticas de fortalecimiento de la educación de ingenieros. El equipo de investigación interdisciplinar indagó sobre los supuestos pedagógicos apropiados para introducir conceptos básicos de programación a estudiantes de ingeniería mecánica, con especial énfasis en: trabajo colaborativo, aprendizaje basado en problemas, motivación inicial, articulación horizontal entre asignaturas y balance adecuado entre conceptualización y procedimientos. Los primeros resultados muestran que la percepción de los estudiantes es positiva y que la secuencia didáctica resultó apropiada para trabajar con ingresantes, lo que confirma el potencial de estas prácticas de taller para aprender programación en el ciclo introductorio de carreras de ingeniería.

**Palabras claves:** Introducción a la Ingeniería, Programación, Robótica, FabLab, Espacios *Maker*, Ingeniería Mecánica, Modalidad taller, Arduino.

### 1 Introducción

Este trabajo presenta un estudio exploratorio en torno a la actividad “Robótica” integrada en la asignatura “Introducción a la Ingeniería Mecánica” en la Facultad de Ciencias Exactas, Ingeniería y Agrimensura (FCEIA) de la Universidad Nacional de Rosario (UNR), Argentina. En el marco de las políticas educativas para el fortalecimiento de la formación de ingenieros [1]–[3], la carrera de Ingeniería Mecánica en el Plan de Estudios 2014 incorporó la modalidad pedagógica de taller físico-virtual en esta actividad curricular introductoria, para realizar prácticas relacionadas con contenidos de primer año [4]. Desde 2020, estas actividades se desarrollan en FabLab UNR [5], un laboratorio construido especialmente para realizar proyectos de diseño, prototipado y manufactura. Este laboratorio, ubicado en la Escuela de Ingeniería Mecánica (EIM) del Centro Universitario Rosario, cuenta con fresadora y torno CNC, impresora 3D, pantógrafo láser, y diversas máquinas pequeñas, herramientas de mano e instru-

mentos de medición. Se cuenta con una gran variedad de equipamiento de electrónica vinculado a la robótica: placas arduino de diversas características, motores paso a paso, bombas de impulsión, sensores diversos, etc. La actividad “Robótica”, se brinda en paralelo con otras actividades, tales como Impresión 3D y Mecanizado CNC.

Las actividades docentes de FabLab UNR se desarrollan en colaboración con el Instituto Rosario de Investigaciones en Ciencias de la Educación (IRICE: CONICET-UNR) a través del equipo de Dispositivos Intermediales Dinámicos [6]. Esto hace de FabLab UNR un entorno de trabajo particular, ya que participan pedagogos e investigadores con base en ciencias humanas y sociales, que contribuyeron a establecer ciertas premisas pedagógicas y motivacionales para el diseño de actividades.

### 1.1 La perspectiva de las ciencias humanas y sociales

Desde un marco socio técnico no determinista, no sólo se asume que la tecnología puede dar forma a la sociedad sino también la sociedad a la tecnología [7], [8]. De este modo, los dispositivos tecnológicos no se conciben como implementaciones cerradas que se replican de manera idéntica en todos los contextos, sino como un saber maleable que también “se deja” modelar por dichos contextos. Esta interacción (acción recíproca) cobra especial relevancia en el análisis de procesos de aprendizaje mediados por tecnología en los que se busca la igualdad de oportunidades.

Una consecuencia esperada de la emergencia de la cultura *DIY (Do It Yourself)*, según algunos autores, es la democratización del conocimiento y de la invención [9]. Sin embargo, la equidad no está garantizada por el acceso físico a los recursos tecnológicos, como puede observarse en las investigaciones sobre brecha digital [9]–[11], sino también por el desarrollo de los procesos superiores de pensamiento que permitan la apropiación. No todos los aprendices son intelectualmente “activos” en su auto-gestión del aprendizaje. Se ha observado, por ejemplo, que algunos estudiantes sólo son “espectadores” de los cursos en video, y en estos casos, se les recomienda realizar actividades durante la observación [12]. Otro comportamiento pasivo es la realización de actividades pero sin reflexionar sobre ellas, por ejemplo, cuando se sigue un video paso a paso imitando cada uno de los pasos para construir un objeto pero sin prestar atención a lo que se está haciendo (sin construir el concepto). La preocupación por la “reflexión” como parte de las actividades *maker* es un tema recurrente en las investigaciones de la comunidad FabLearn [13].

Fuera de los espacios institucionales, los videos que prometen hacer hasta lo más complejo en pocos minutos, promueven una cultura *maker* para la creación de artefactos sin intervención de la comprensión ni la creatividad. Es por ello que interesa distinguir entre la “cultura maker” como movimiento global de aprendizaje informal y los “espacios maker” que han tenido lugar a nivel internacional en los ambientes educativos formales.

Los *FabLab* son laboratorios de fabricación en contextos educativos [14]. A medida que se desarrollan estos laboratorios, emerge la comunidad *FabLearn* conformada por los investigadores y educadores a nivel internacional que estudian la evolución de los *FabLab* con el objetivo de mejorar las prácticas [15]–[18].

Los fundamentos teóricos de los espacios *maker* se remontan al constructivismo [19], [20], al construccionismo de Papert [21], a la pedagogía crítica [22], [23] y al aprendizaje basado en proyectos [24]. Pero dichas raíces teóricas no fueron suficientes para la emergencia de estos nuevos talleres. La novedad radica en que los espacios *maker* utilizan tecnologías de reciente aparición -como la impresión 3D, el maquetado láser, las placas Arduino en robótica- que confluyen con plataformas de software que facilitaron el acercamiento del estudiante al diseño y la programación, tales como el lenguaje Logo [25], o sus sucesores Scratch [26], [27], y NetLogo [28].

En los contextos educativos, se espera que los FabLab estén abocados a transmitir conceptos (apoyados en una práctica concreta), con la guía de profesionales expertos en cada disciplina y en relación con otras asignaturas de la carrera, sin perder su carácter de taller.

La idiosincrasia y los objetivos de cada carrera se complementan de modos diversos con la iniciativa del pensamiento computacional. No es lo mismo aprender a programar con el objetivo en mente de transformarse en un diseñador de video juegos, que aprender a programar para componer música o aprender a programar para ser ingeniero.

Cada año, los alumnos ingresantes a la carrera de Ingeniería Mecánica UNR escriben sus biografías en una actividad solicitada por el equipo de profesores de Introducción a la Ingeniería Mecánica. En estas biografías vuelcan sus intereses, inquietudes, la historia de por qué han elegido esa carrera. Si bien el imaginario del ingresante a ingeniería mecánica suele incluir expectativas poco realistas sobre las posibilidades de la profesión, es claro que éstas giran en torno a la manipulación de componentes físicos y dispositivos móviles. Si bien se realizó una experiencia previa de programación con lenguajes visuales [29], el equipo llegó a la conclusión de que la programación con placas Arduino de un robot con funcionalidad simple, es más apropiada para ingeniería mecánica, por la cercanía entre las primitivas y los componentes mecánicos.

## 1.2 El profesor acompaña a construir el concepto, no sólo el objeto

Como se mencionó antes, la preocupación de los profesores por dar lugar a la reflexión dentro de las prácticas *maker* aparece en numerosas investigaciones sobre espacios *maker*. Sin embargo, no hay un consenso claro acerca de qué procesos específicos intervienen en la reflexión [13]. Lo que constituye un aporte de nuestra investigación es que agregamos a los enfoques constructivistas elementos teóricos de base cognitiva para explicar la importancia de introducir el contenido conceptual antes de la automatización del contenido procedural. Lo “conceptual”, también llamado contenido semántico, o declarativo, desde un punto de vista cognitivo, se equipara con “aquello en lo que hay que detenerse a reflexionar” desde la óptica de los enfoques constructivistas.

Desde los inicios de la iniciativa para la promoción del pensamiento computacional se estableció claramente que no se trataba de transmitir únicamente habilidades sino conceptos [30], [31]. Ambos aspectos son componentes de una competencia [32]. Esta afirmación que parece estar clara a nivel teórico, en la práctica *maker*, donde la

balanza se inclina hacia lo procedimental, se está produciendo lentamente una relegación de lo conceptual a un segundo plano.

El desafío en el diseño de esta consigna fue decidir en qué etapa de la actividad introducir los conceptos sin obstaculizar el procedimiento. Para explicar la importancia de esta secuencia, utilizaremos como analogía el proceso de aprender a andar en bicicleta. Cuando una persona sabe andar en bicicleta no necesita prestar atención a los movimientos de sus pies sobre los pedales. De hecho, si le pidiéramos que tome conciencia de cómo va pedaleando podría incluso caerse por observar sus pies para explicitar algo ya inconsciente. Sin embargo, en su etapa de aprendizaje en la niñez probablemente habrá seguido instrucciones y habrá pensado detenidamente dónde colocar los pies y cómo mantener el equilibrio. Los consejos recibidos en ese momento suelen ser bienvenidos y captados con atención. Se dice que el conocimiento está en un estadio *declarativo* [33]. A medida que se repite la misma práctica, el conocimiento se vuelve procedural y automático, adquiere una especie de inercia. Una vez automatizado el saber andar en bicicleta, es difícil volver a hacer explícitos los primeros pedaleos.

La analogía permite vislumbrar que cuando los aprendices de videos expeditivos ya han automatizado la práctica de ensamblar, subir programas desde Internet a un robot y verlo funcionar sin necesidad de implicar su propio razonamiento lógico, probablemente se resistan a volver sobre los primeros pasos para cuestionarse cómo está escrito el código, para qué sirve cada instrucción, o cuál es el concepto de variable. Cuando surjan dificultades irresolubles por desconocer el origen de los errores, es probable que pierdan la motivación y abandonen la tarea. Adquirir un automatismo no garantiza que el aprendizaje adquirido sea correcto. Todo aprendizaje procedural conlleva el riesgo de automatizar algo equivocado. La mejor oportunidad que tienen los profesores de introducir conceptos durante actividades procedimentales como las que se enseñan en los FabLab, es *antes de la automatización*, cuando la actividad está en su *etapa declarativa*, es decir, en los primeros pasos del aprendizaje.

El conocimiento declarativo es verbal. Se expresa en palabras. Esto también es particularmente importante en un ambiente *maker* porque existe la posibilidad de trabajar sin hablar. Un modo de activar la explicitación verbal es intentar resolver problemas en equipo, otro modo es que el docente explicita claramente el nombre del término y su definición. Investigaciones recientes sobre aprendizaje de conceptos en diseños instruccionales revelan que una forma efectiva de enseñar un concepto es mostrar su definición de manera contigua a los ejemplos [34]–[36]. En contraste con la típica separación entre teoría y práctica, esto indicaría que no debería separarse demasiado el componente teórico del concepto (su definición), del componente concreto (el ejemplo).

En vez de pensar en el taller como un lugar para poner en práctica la teoría del aula, sería más productivo organizar la práctica de modo tal que las definiciones se den durante el procedimiento en el momento preciso en que surge el ejemplo. Por ejemplo, en la experiencia *maker* de robótica, cuando se desea encender un motor, debe cambiarse el valor de la variable. Es un buen momento para presentar el concepto de

variable informática<sup>1</sup> con una definición breve: “un espacio de memoria en el que se almacena un valor que puede ser leído o modificado”.

### 1.3 Motivación

Otro problema que surge al diseñar una actividad breve con un objetivo complejo es cómo mantener la motivación en la tarea. Según Deci y Ryan [37], la motivación intrínseca, está basada en tres necesidades psicológicas: (1) curiosidad, (2) percepción de causación personal (de ser el autor del resultado) y (3) percepción de efectividad (que el resultado sea adecuado). No es sencillo lograr satisfacer cada una de estas necesidades durante actividades de enseñanza porque la relación entre ellas puede dar lugar a un desequilibrio.

Por ejemplo, la percepción de causación personal se puede lograr permitiendo que los alumnos trabajen solos. Pero la percepción de efectividad (3) sólo se obtiene cuando el resultado es correcto: el artefacto construido funciona. Para evitar que los alumnos se frustren por no lograr hacer funcionar el dispositivo, interviene el profesor, pero su intervención no debe ser tan abrumadora que el estudiante no perciba su propia causación personal y sienta que lo construyó el profesor.

Por otra parte, para que algo despierte curiosidad, no debe ser totalmente conocido ni totalmente desconocido. Un criterio para que la actividad no fuera totalmente conocida fue proponer instrucciones no típicas, es decir, que no pudieran hallarse en Internet fácilmente. Para que la actividad fuera parcialmente conocida (lo que también estimula la participación) se propuso a los alumnos que eligieran ellos mismos en qué actividades de FabLab inscribirse. Evidentemente, las que más les interesen deben ser conocidas parcialmente, y esta libre decisión también impacta positivamente sobre la sensación de causación personal.

### 1.4 Contenidos

Al equilibrio entre aspectos cognitivos y motivacionales se sumaba el requisito de transmitir contenidos. El objetivo principal era transmitir conceptos de programación a estudiantes de ingeniería mecánica vinculados al área específica, pero con miras a una recuperación futura de dichos contenidos en la asignatura Informática. Los objetivos secundarios eran: (a) lograr un primer acercamiento con los conceptos de robótica y de programación, (b) reflexionar sobre temas trascendentes para el desarrollo de la ingeniería mecánica como la predicción de eventos en un contexto controlado, (c) experimentar con un proceso de ajuste de variables, (d) reflexionar sobre la precisión de las mediciones indirectas, (e) aplicar procesos de validación e iteraciones de mejora, (f) aprender en un entorno colaborativo.

Para lograr introducir estos conceptos en la fase declarativa del aprendizaje de la construcción de un robot, se debía planificar anticipadamente en qué momento de la

---

<sup>1</sup> Se aclara “variable informática” para distinguirlo del concepto de variable matemática, ya conocido por los alumnos.

actividad se introduciría cada uno, aprovechando las cuestiones que surgirían durante la programación del robot.

## **2 Método**

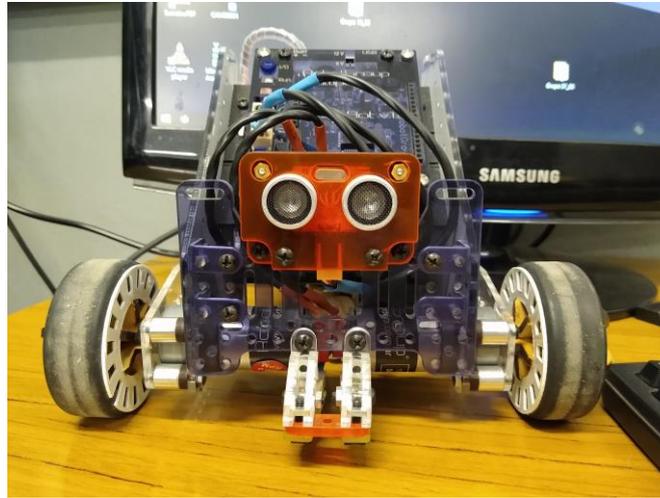
El abordaje metodológico fue de tipo exploratorio, con características de investigación acción ya que se intentó producir conocimiento durante el desarrollo de la actividad “Robótica”. Los encuentros se repitieron semanalmente con diferentes grupos de alumnos. Se analizó la experiencia de cada encuentro y se realizaron ajustes basados en ese análisis.

El equipo de investigación estuvo integrado por un doctor en Ingeniería, una doctoranda en Ingeniería -ambos docentes de la asignatura Introducción a la Ingeniería Mecánica-, y una doctoranda en Psicología, que observó el desarrollo de la actividad y analizó los datos recabados.

Las variables indagadas fueron la percepción de los estudiantes de los diversos aspectos de la experiencia, como la claridad en la consigna y en la metodología de trabajo, la adecuación del espacio físico y de las herramientas e instrumentos. Se indagó cuáles fueron los contenidos de otras asignaturas que los estudiantes pudieron integrar en la experiencia, los criterios funcionales utilizados en la resolución del problema, la relación con otros estudiantes y con los docentes, y las sugerencias para próximas actividades.

## **3 Experiencias y observaciones**

Se utilizó el robot educativo Múltiple N6 diseñado en Argentina por RobotGroup [38] (Figura 1) que consta de dos motores independientes de corriente continua para dos ruedas delanteras, una rueda trasera más pequeña con movimiento libre, dos sensores infrarrojos reflexivos para medir luz y un sensor ultrasónico para medir distancia.



**Fig. 1.** Robot Multiplo N6 de RobotGroup ensamblado.

El kit contiene un controlador DuinoBot compatible con Arduino programable con Arduino Ide o con Minibloq, se utilizó éste último por poseer interfaz gráfica y posibilidad de observar el código generado a la vez [39].

La actividad tenía una duración de 3 horas. Se reprodujo en varias oportunidades con diferentes grupos que no superaron los seis participantes hasta cubrir un total de 51 estudiantes que se inscribieron libremente para participar. Las opciones eran Robótica, Impresión 3D, Maquinado CNC, Estructuras estáticas y Electro neumática.

La sala de trabajo contaba con un pizarrón móvil, una mesa central y cuatro bancos alrededor con una disposición que permitía la conversación de frente, una segunda mesa con una computadora y un banco alargado frente a ella con capacidad para tres alumnos y un espacio para que otros estuvieran de pie. Se contaba con un espacio libre de 15 metros cuadrados para la prueba de funcionamiento del robot. El desplazamiento por los espacios era funcional a las tareas.

Las cuatro consignas de programación eran básicas y se ordenaban desde la más simple a la más compleja. La consigna se comentaba primero en la mesa de charla. Cuando una idea surgida era importante, se anotaba en el pizarrón. Cada consigna se problematizaba oralmente en equipo, luego se escribía el *pseudo código* en el pizarrón y sólo cuando era evidente que los estudiantes habían comprendido la lógica de las instrucciones, se pasaba a la computadora para programar en Minibloq (Figura 2).



**Fig. 2.** Secuencia de producción previa a la prueba del robot: (1) Mesa de charla, (2) pizarrón de conceptos, (3) programación en Minibloq.

Algunos participantes escribieron código, otros colaboraron aportando ideas sobre cada instrucción, pero este orden era flexible, porque el grupo estaba en constante movimiento, una vez finalizado el programa, se subía el código al robot, se trasladaba a la zona libre para probarlo (Figura 3) y el equipo se reorganizaba en la siguiente tarea de programación. Se trató de evitar que las mismas personas programaran todas las consignas.



**Fig. 3.** Zona de prueba del robot luego de la programación.

Durante las pruebas en el piso se analizaban las diferencias entre lo pensado y el resultado, y se actuaba en consecuencia indagando las causas de error o proponiendo modificaciones.

La secuencia didáctica (Tabla 1) se planificó de manera tal de integrar los conceptos que se buscaba transmitir dentro de las tareas de programación del robot.

**Tabla 1.** Secuencia didáctica de la actividad.

Orden	Duración	Descripción	Materiales
1	10-15'	<b>Presentación de los integrantes</b> del equipo con sus experiencias y saberes en general.	
2	10'-15'	<b>Presentación de la actividad</b> <b>1.</b> Delinear la modalidad del espacio de trabajo: sugerencias para trabajar en equipo. <b>2.</b> Introducir los conceptos de robótica y programación. <b>3.</b> Reflexionar sobre posibles usos de los robots en ingeniería. <b>4.</b> Presentar la Interfaz de programación.	Consigna escrita describiendo la actividad. Programa Minibloq (programación en bloques).
3	135'	<b>Práctica</b> <b>1.</b> Experimentar con el movimiento del robot dentro de una línea recta. ( <i>Consigna 1: Mover el robot en línea recta por un tiempo</i> ). <b>2.</b> Probar distintas velocidades. <b>3.</b> Observar otras dificultades ( <i>Consigna 2: Realizar ajuste para que ambas ruedas frenen al mismo tiempo</i> ). <b>4.</b> <i>Consigna 3: Programar y correr un algoritmo para avanzar hasta un obstáculo y marcar el momento de encuentro (giro 180 grados)</i> . <b>5.</b> Intentar realizar una medición indirecta a partir de la detección de la línea. Reflexionar sobre qué papel juega el orden de la secuencia y las estructuras de control en el programa. <b>6.</b> <i>Consigna 4: Programar un algoritmo para que el robot siga una línea.</i>	Robot Múltiple N6 de RobotGroup ensamblado en FabLab UNR (Figura 1).

4	15'	<b>Cierre de la actividad:</b> Reflexión final. Puesta en común de la actividad. Reflexionar acerca de otras aplicaciones posibles de este tipo de tecnología, pensar como fue el desarrollo de la actividad, que aprendieron, etc. Encuesta sobre la actividad para realizar en sus casas.	
---	-----	---	--

### Presentación de los integrantes del equipo

Cuando llegaron los participantes al laboratorio pasaron a la sala, se sentaron alrededor de la mesa de charla y se presentaron. Se les pidió que contaran brevemente los motivos para elegir ingeniería mecánica como carrera y para elegir la actividad de robótica en particular. Los profesores se presentaron y hablaron de su motivación hacia la robótica.

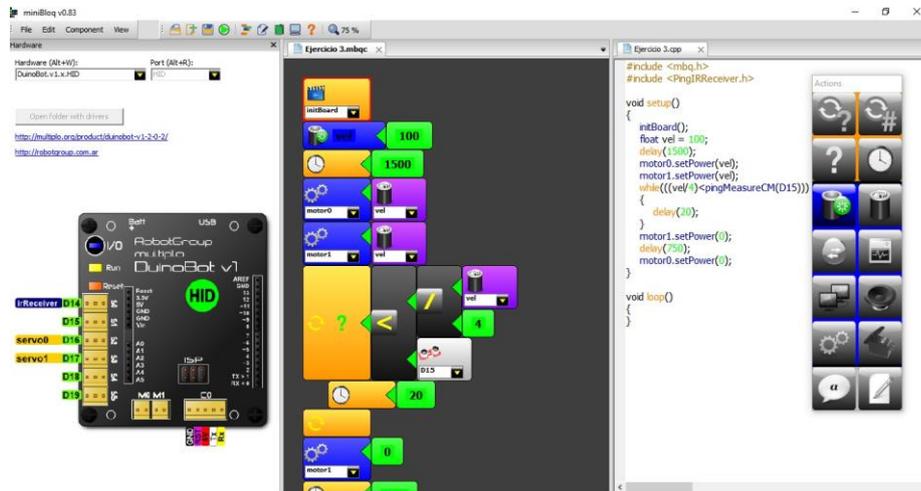
### Presentación de la actividad

*Delinear la modalidad del espacio de trabajo: sugerencias para trabajar en equipo.* Luego de las presentaciones el profesor explicó cómo se desarrollaría la actividad de programación del robot en equipo. Mencionó ciertas recomendaciones para la comunicación: (a) La opinión de cada miembro del equipo es igualmente valiosa. (b) La idea que aporta una única persona puede no ser suficiente para resolver un problema complejo pero, a veces, se potencia con el aporte de otra persona, por eso es importante que todos participen. (c) Si bien todos son libres de participar, para poder escuchar a todos es necesario respetar los turnos en la conversación.

*Introducir los conceptos de robótica y programación.* Se intentó introducir los conceptos de robótica y de programación a partir de los conocimientos previos de los estudiantes. Para ello los profesores preguntaron al grupo qué idea tenían de *robot*, si conocían ejemplos de robots y cuáles eran características que hacen que un dispositivo sea considerado un robot. A medida que los alumnos participaban aportando ejemplos de robots que ellos conocían, se hacían preguntas acerca de sus características y se intentó ir rescatando aquellos caracteres definitorios de un robot y anotándolos en un pizarrón, a saber: (a) *Movimiento*: Puede producir movimiento a través de motores o actuadores. (b) *Autonomía*: puede realizar ciertas tareas sin la intervención directa del usuario. (c) *Programa*: Puede ser relativamente autónomo porque tiene un programa, es decir, un conjunto de instrucciones. (d) *Sensores* para captar información que ingresa al programa para determinar qué acciones realizar. (e) *Comunicación con el entorno*: los sensores “ingresan” información al programa y los motores son activados por instrucciones que “salen” del programa.

*Presentación de la interfaz de programación.* Se presentó Minibloq, una plataforma visual de programación para Arduino y otros dispositivos de robótica que permite usar bloques en lugar de líneas de código (Figura 4). No obstante, cuenta con un gene-

rador de código que permite a los estudiantes observar en tiempo real cómo se escriben las instrucciones para cada bloque.



**Fig. 4.** Plataforma Minibloq: A la izquierda, Vista de Hardware; en el centro, editor de Minibloq; a la derecha, editor de código.

En esta etapa del encuentro, los participantes se trasladaron hacia la pantalla del ordenador ubicado cerca de la mesa de charla. Se presentó el entorno de Minibloq, las tres vistas, y las herramientas. Se recomendó ubicar el mouse sobre los distintos componentes del panel para que se active la leyenda que indica para qué se usa cada componente.

### Fase de práctica

*Experimentar con el movimiento del robot dentro de una línea recta.* En primer lugar se trató de identificar con qué dispositivos se contaba, cuál era el nombre de cada uno (se pidió a los estudiantes que observaran los nombres escritos en la placa) y cuáles eran sus características. Se identificaron los tres sensores, los dos motores y el programa. Los sensores que están en la parte inferior del robot FabLab son sensores infrarrojos que miden el color (la cantidad de luz) de la superficie. Los sensores de proximidad miden la distancia al objeto más próximo frente al robot. Se comentó para qué otros usos sirven estas placas. Se mostró como la placa se enchufa para subir el programa que se realiza en la PC con Minibloq.

La primera consigna fue Mover el robot en línea recta por un tiempo. Se preguntó a los estudiantes cómo creían que se movía el robot para comenzar a pensar cómo programarla. Con “línea recta” no referimos a evitar las desviaciones sino simplemente activar los motores por unos segundos. Esto se programa en Minibloq activando los motores motor1 y motor2 y agregando una demora de unos pocos segundos antes de apagarlos.

Aprovechando la consigna se explicó el concepto de *variable* asociado a los sensores y motores. El profesor explicó que cada motor o sensor posee un espacio en el cual se guarda un número que puede conocerse o modificarse. Se escribió en el pizarrón la forma habitual de una asignación de variables. Por ejemplo, si  $MI$  es la variable del primer motor,  $MI = 50$  estaría asignando el 50 por ciento de potencia al motor para que encienda a una velocidad media. Los valores de potencia asignados a un motor son porcentajes que van de 0 (apagado) a 100 (velocidad máxima posible para ese motor), es decir,  $MI = 0$  significa apagar el motor (encender es asignar un número distinto de 0). En el programa simplemente se debe dar clic en el panel sobre el símbolo de motor (engranajes en la versión 0.82 Beta), y asignarle un valor de potencia.

El sensor de proximidad tendría asociada la variable  $S1$ , donde se almacena un número que indica la distancia hasta el objeto más próximo calculada por ultrasonido. Para que el robot realice alguna acción dependiendo de la información que proporciona la variable del sensor  $S1$ , es necesario especificarlo en el programa. Comúnmente se establece la condición de que si detecta que la distancia próxima es menor que un valor mínimo (por ejemplo, si se acerca a una pared, o si alguien coloca una mano enfrente del robot), el robot se detiene.

*Probar distintas velocidades.* Se probó cambiar en el programa la potencia de los motores y observar el recorrido real del robot. También se planteó a los alumnos qué efecto tendría en el robot si una rueda fuera más rápida que otra. El efecto es que el robot daría un giro. Dado que la consigna 1 indicaba que el robot fuera en línea recta por un tiempo, en ese caso, ambas ruedas debían ir a la par, a la misma velocidad.

*Observar otras dificultades.* El robot de FabLab, realiza una pequeña desviación no deseada (giro corto) al frenar. Este hecho observado en la consigna 1, se utilizó para desarrollar la consigna 2: *Realizar ajuste para que ambas ruedas frenen al mismo tiempo.* Se planteó a los estudiantes el problema y se aprovechó esta instancia para introducir nociones de *predicción* y *control*. En primer lugar, se preguntó cómo es posible averiguar qué es lo que ocasiona ese giro. Se puede comenzar descartando de a una las *posibles causas*, por ejemplo, averiguar si se trata de algún desperfecto en el piso. Se puede trasladar el robot a otro tipo de piso dentro del establecimiento (variar las condiciones) y observar nuevamente la trayectoria. Si el error persiste, se descarta que la causa sea el piso. Luego de descartar otras posibilidades, se llegó a la conclusión de que podría tratarse de un inconveniente mecánico que hace que uno de los motores funcione por más tiempo. Se preguntó a los estudiantes cómo corregir el programa para que el motor que se detenía antes, funcionara unas décimas de segundos más (Figura 5).

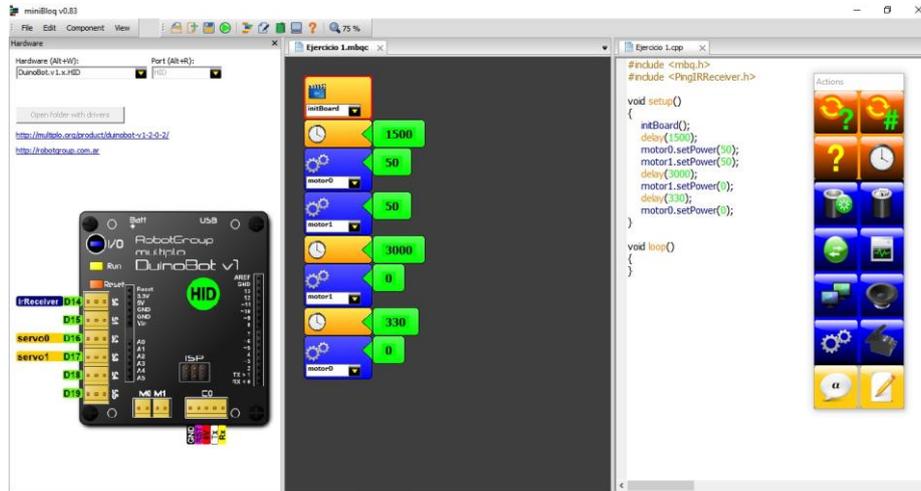


Fig. 5. Código en Minibloq para encendido de motores y retraso de una rueda.

Como en otras instancias, para estimular la participación se esperó que los estudiantes realicen sugerencias antes de que las planteen los profesores. Los estudiantes propusieron varias formas de medir la diferencia entre las ruedas, filmarla para medir el tiempo en el video, medir la diferencia en el espacio para averiguar el tiempo de ese recorrido, entre otras.

*Consigna 3: Programar y correr un algoritmo para avanzar hasta un obstáculo y marcar el momento de encuentro (giro 90°).* En esta consigna se utiliza el sensor ultrasónico para detectar la cercanía de objetos. El robot debe detenerse cuando encuentre un objeto en el frente a una distancia mínima a establecer por programa. En este momento se introducen los *conceptos de iteración* (bucle) y *condición* para que el sensor repita la lectura de la distancia hasta que esa distancia sea menor o igual a la mínima establecida.

Anticipando la siguiente consigna, se explicó a los estudiantes que el sensor infrarrojo que está debajo del robot, funciona emitiendo luz con un diodo emisor de infrarrojo y recibiendo luz por rebote con un fototransistor. Si lo que está “observando” es una superficie oscura, la superficie absorbe mayor cantidad de luz y rebota menos luz hacia el fototransistor. Cuando la *variable* del sensor infrarrojo indicara un valor bajo, podría estar frente a una línea. Se colocó una cinta aisladora negra para dibujar la línea en el piso.

Dado que el programa anterior tenía una lógica similar que incluía una iteración para la detección de obstáculos, se reutilizó cambiando el sensor y la condición de fin para la detección de una línea en el piso.

*Consigna 4: Programar un algoritmo para que el robot siga una línea.* En esta consigna se complejiza más el uso de las condiciones. Si el robot comienza el recorrido desde una línea, se debe chequear si ambos sensores están sobre la línea o, caso contrario, averiguar cuál de los dos sensores se movió hacia afuera para volver a posicionarlo en la línea.

En este punto, se puede reflexionar sobre qué papel juega el orden de la secuencia y las estructuras de control en el programa, así como observar la complejidad de las estructuras de condición anidadas.

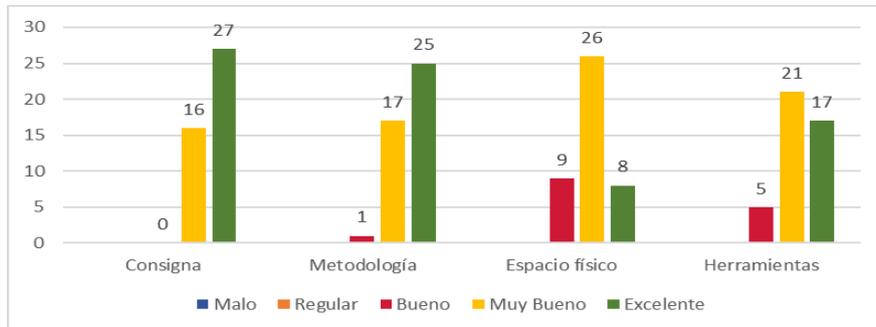
### 3.1 Primeros resultados en torno a las percepciones generales

Al finalizar la actividad, los participantes completaron desde sus casas una encuesta acerca de sus impresiones sobre el encuentro basada en las variables que se mencionaron en la sección *Método* (Tabla 2).

**Tabla 2.** Preguntas sobre la actividad de robótica.

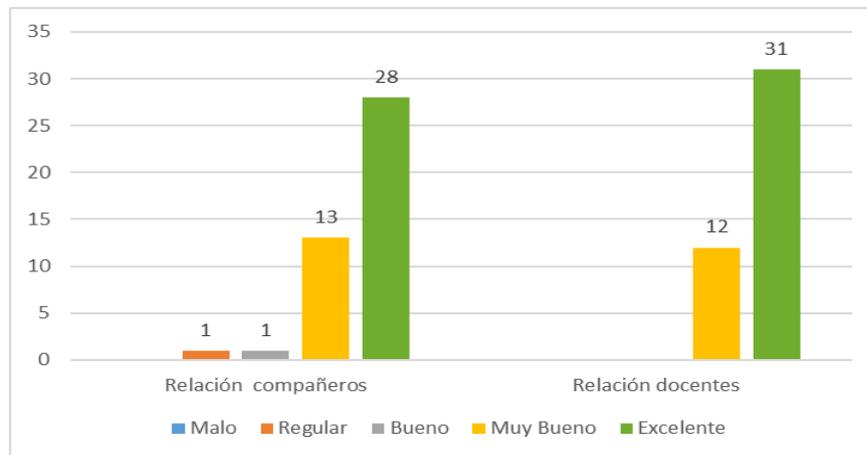
1	¿Cuál experiencia <i>maker</i> realizó?
2	¿La consigna propuesta fue clara?
3	¿La metodología de trabajo fue clara?
4	¿El espacio físico piensa que fue adecuado?
5	¿Qué opinión tiene en torno a los elementos utilizados (herramientas, materias primas, instrumentos, etc.)?
6	¿Cuáles conceptos de otras asignaturas pudieron integrar en esta experiencia?
7	¿Qué habilidades creativas pudo poner en juego en esta experiencia?
8	¿Qué criterios funcionales propuso en la resolución del problema?
9	¿Cómo fue la relación con sus compañeros en el marco de la experiencia?
10	¿Qué nos sugieren para una óptima conformación del grupo?
11	¿Cómo fue La relación con los docentes en el marco de la experiencia?

Las preguntas 2 a 5 hacen referencia a la percepción general del encuentro en relación a la consigna, metodología, espacio y herramientas. Las preguntas 9 y 11 también indagan la percepción general pero vinculada a las relaciones personales con compañeros y docentes. El grado de acuerdo del alumno en todas las preguntas sobre percepción se expresó con una escala de 1 a 5. Por otro lado, las preguntas 6 a 8 son metacognitivas, implican una reflexión del estudiante acerca de su propio proceso de aprendizaje en relación a la carrera, la pregunta 6 en particular refiere a los conocimientos previos. Estos tres ítems eran preguntas abiertas para habilitar una mayor especificidad en la respuesta. En la Figura 6 se exponen los resultados de las preguntas 2 a 5.



**Fig. 6.** Resultados sobre percepción de Consigna, Metodología de trabajo, Espacio físico y Herramientas utilizadas

Sobre el total de respuestas en torno a la percepción se observa que ningún estudiante calificó con la categoría mala o regular. Los resultados mejor calificados estuvieron en torno a la Consigna y Metodología de trabajo, mientras que el Espacio físico y Herramientas utilizadas obtuvieron una puntuación menor. Entendemos que esto se debe por una parte, a que el espacio físico utilizado, todavía requiere ajustes por una mudanza retrasada, y en torno a las herramientas el equipo manipulado es antiguo y existen robots más modernos, que los mismos estudiantes señalaron. En la Figura 7 se exponen los resultados de las preguntas 9 y 11.



**Fig. 7.** Percepción de la relación con compañeros y con docentes.

Las respuestas reflejan el buen clima de trabajo entre compañeros y con los docentes, la buena disposición a colaborar y el respeto mutuo durante la actividad.

## 4 Conclusiones preliminares

Los resultados de las preguntas vinculadas a la percepción general de la experiencia de robótica muestran que la percepción fue positiva. Esto se evidenció durante los encuentros por la participación activa y la voluntad de colaborar en todas las consignas. Los programas en Minibloq tuvieron variaciones en cada uno de los encuentros ya que las alternativas de solución se iban generando de acuerdo a las suposiciones realizadas por cada grupo y a las decisiones tomadas.

Advertimos que en relación a la adquisición de conceptos habría sido productivo compartir en el campus un resumen de la experiencia explicitando los conceptos aprendidos y los programas generados por cada grupo, luego del cierre de la actividad, aunque estimamos conveniente no dar a conocer públicamente todas las actividades para mantener el efecto de novedad que sostiene la curiosidad (y la motivación) durante el encuentro.

No analizamos aquí las preguntas abiertas acerca de la vinculación con conceptos de otras asignaturas pero en relación a la percepción general deducimos que la secuencia didáctica resultó apropiada para trabajar con ingresantes, lo que confirma el potencial de estas prácticas de taller para aprender programación en el ciclo introductorio de ingeniería mecánica.

## 5 Referencias

1. CONFEDI, «Propuesta de estándares de segunda generación para la acreditación de carreras de ingeniería en la República Argentina», *Aprob. Por Asam. Cons. Fed. Decanos Ing. Repúb. Argent. Rosario*, vol. 1, 2018.
2. K. C. Ferrando, «Perfil del Ingeniero y formación complementaria en carreras de Ingeniería», 2014.
3. P. V. Paoloni, A. C. Chiecher, y R. C. Elisondo, «Graduados de ingeniería y competencias genéricas. Cinco estudios de la última década que recuperan sus valoraciones y experiencias», *Rev. Educ. En Ing.*, vol. 14, n.º 28, pp. 54-64, 2019.
4. G. Rodríguez, M. Raposo, F. Sklate, y P. Demartini, «Análisis de experiencias de una cátedra de Introducción a la Ingeniería Mecánica», p. 11, 2018.
5. «FabLab UNR», *FabLabs.io - The Fab Lab Network*, 2022. <https://www.fablabs.io/labs/fablabunr> (accedido 13 de junio de 2022).
6. P. S. San Martín, «Dispositivos Hipermediales Dinámicos», p. 5, 2011.
7. D. MacKenzie y J. Wajcman, *The social shaping of technology*. Buckingham, UK: Open University Press, 1999. Accedido: 28 de abril de 2022. [En línea]. Disponible en: <http://mcgraw-hill.co.uk/openup/>
8. P. S. San Martín, «Aspectos sociales y tecnológicos del Dispositivo Hipermedial Dinámico desarrollados en diferentes contextos educativos», *Rev. Educ.*, vol. 5, n.º 5, pp. 81-98, 2013.
9. J. van Dijk, *The Digital Divide*. John Wiley & Sons, 2020.
10. J. A. Van Dijk, «Digital divide research, achievements and shortcomings», *Poetics*, vol. 34, n.º 4-5, pp. 221-235, 2006.
11. J. A. Van Dijk, «Digital divide: Impact of access», *Int. Encycl. Media Eff.*, pp. 1-11, 2017.

12. K. Koedinger, E. McLaughlin, J. Kim, J. Jia, y N. Bier, «Learning is Not a Spectator Sport: Doing is Better than Watching for Learning from a MOOC», pp. 111-120, mar. 2015, doi: 10.1145/2724660.2724681.
13. G. E. Baykal, M. Van Mechelen, M.-L. Wagner, y E. Eriksson, «What FabLearn talks about when talking about reflection—A systematic literature review», *Int. J. Child-Comput. Interact.*, vol. 28, p. 100256, 2021.
14. P. Blikstein y D. Krannich, «The makers' movement and FabLabs in education: experiences, technologies, and research», en *Proceedings of the 12th international conference on interaction design and children*, 2013, pp. 613-616.
15. D. Scaradozzi, L. Guasti, M. Di Stasio, B. Miotti, A. Monteriù, y P. Blikstein, *Makers at School, Educational Robotics and Innovative Learning Environments: Research and Experiences from FabLearn Italy 2019, in the Italian Schools and Beyond*. Springer Nature, 2021.
16. P. Blikstein, «Maker movement in education: History and prospects», *Handb. Technol. Educ.*, vol. 419, p. 437, 2018.
17. C. Fernandez, T. Hochgreb-Haegele, y P. Blikstein, «From “Playful Activities” to “Knowledge Building”: A Case Study about a Teacher's Perceptions on the Role of Experiments», 2021.
18. J. A. Valente y P. Blikstein, «Maker education: Where is the knowledge construction?», *Constr. Found.*, vol. 14, n.º 3, pp. 252-262, 2019.
19. E. Von Glasersfeld, *Radical constructivism: a way of knowing and learning*. London ; Washington, D.C: Falmer Press, 1995.
20. E. Von Glasersfeld, «An introduction to radical constructivism», *Inven. Real.*, vol. 1740, p. 28, 1984.
21. S. Papert y I. Harel, «Situating constructionism», *Constructionism*, vol. 36, n.º 2, pp. 1-11, 1991.
22. P. Freire, *Pedagogía del oprimido*. Siglo xxi, 2005.
23. I. Illich, «La sociedad desescolarizada». Ediciones Godot Buenos Aires, Argentina, 2011.
24. D. Kokotsaki, V. Menzies, y A. Wiggins, «Project-based learning: A review of the literature», *Improv. Sch.*, vol. 19, n.º 3, pp. 267-277, nov. 2016, doi: 10.1177/1365480216659733.
25. S. Papert, «What is Logo? Who needs it», en *Logo philosophy and implementation*, Logo Computer Syst., Inc., 1999, pp. 4-16.
26. M. Resnick *et al.*, «Scratch: programming for all», *Commun. ACM*, vol. 52, n.º 11, pp. 60-67, 2009.
27. J. Maloney, M. Resnick, N. Rusk, B. Silverman, y E. Eastmond, «The scratch programming language and environment», *ACM Trans. Comput. Educ. TOCE*, vol. 10, n.º 4, pp. 1-15, 2010.
28. U. Wilensky, «NetLogo. Evanston, IL: Center for connected learning and computer-based modeling, Northwestern University». 1999.
29. N. G. Monjelat, G. L. Rodríguez, y P. S. S. S. Martín, «Modelado y simulación de un sistema mecánico simple: Programar en primer año de ingeniería», *Rev. Educ. En Ing.*, vol. 13, n.º 25, Art. n.º 25, feb. 2018, doi: 10.26507/rei.v13n25.796.
30. J. Wing, «Computational thinking», *J. Comput. Sci. Coll.*, vol. 24, n.º 6, pp. 6-7, jun. 2009.
31. J. M. Wing, «Computational thinking», *Commun. ACM*, vol. 49, n.º 3, pp. 33-35, mar. 2006, doi: 10.1145/1118178.1118215.
32. M. Rodríguez Moneo, «El proceso de enseñanza y aprendizaje de competencias», en *Evaluación global de los resultados del aprendizaje en las titulaciones dentro del Espacio Europeo de Educación Superior*, 2011, pp. 19-44.

33. J. R. Anderson, «Acquisition of cognitive skill.», *Psychol. Rev.*, vol. 89, n.º 4, p. 369, 1982.
34. K. A. Rawson, R. C. Thomas, y L. L. Jacoby, «The Power of Examples: Illustrative Examples Enhance Conceptual Learning of Declarative Concepts», *Educ. Psychol. Rev.*, vol. 27, n.º 3, pp. 483-504, sep. 2015, doi: 10.1007/s10648-014-9273-3.
35. A. Jamrozik y D. Gentner, «Relational labeling unlocks inert knowledge», *Cognition*, vol. 196, p. 104146, mar. 2020, doi: 10.1016/j.cognition.2019.104146.
36. V. D'Angelo y M. Trench, «Defending Diversity: Providing Examples from Different Domains Enhances Application of System Principles Beyond the Domains Covered by the Examples.», presentado en Cognitive Science Society «Cogsci 2022: Diversity & Inclusion» (in press), Toronto, Ontario, jul. 2022.
37. E. L. Deci y R. M. Ryan, «Cognitive evaluation theory», en *Intrinsic motivation and self-determination in human behavior*, Springer, 1985, pp. 43-85.
38. «robotgroup.com.ar». <http://www.robotgroup.com.ar/web/> (accedido 15 de junio de 2022).
39. «miniBloq». <http://blog.minibloq.org/> (accedido 15 de junio de 2022).