



Aprendiendo a programar con el Robot Educativo Programable (REP)

Estudio de caso: Introducción a la Programación Imperativa, asignatura perteneciente a las carreras de informática de la UNNOBA.

Paula Lucrecia Lencina

Director: Dra. Russo, Claudia.

Co-Director: Dra. De Giusti, Laura.

Tesis presentada para obtener el grado de Magister en Tecnología Informática Aplicada en Educación

Facultad de Informática - Universidad Nacional de La Plata

Abril de 2020

DEDICATORIA.

A mis hijas Sarah y Pilar, a mi hijo Antonio; quienes sin dudas son la fuente de inspiración y de motivación para continuar adelante en aquellos momentos de debilidad y dificultad.

Y a mi compañero de vida Pablo, por su apoyo y comprensión, y por las ausencias soportadas.

AGRADECIMIENTOS.

A mi familia,

Especialmente a mi madre Graciela y mi padre Juan, por el amor y contención brindados en mi formación personal y profesional, y por el sacrificio puesto en todos estos años de formación académica.

A la prestigiosa UNNOBA por haberme permitido formar parte de ella

Que gracias a su cuerpo docente hoy dispongo de todos los conocimientos, herramientas, habilidades y valores para poder desenvolverme con éxito en mi vida laboral.

A quienes dirigieron mi trabajo de Tesis

Quienes supieron conducirme y guiarme con sus saberes y experiencias a lo largo de todas las etapas del proceso de investigación. Gracias por asumir con responsabilidad y entrega esta tarea.

Y a todos aquellos que fueron partícipes de esta experiencia.

Por acompañarme en este hermoso recorrido ofreciéndome su apoyo incondicional.

Sin cada uno de ustedes no hubiera sido posible alcanzar este sueño. A todos un eterno

GRACIAS

ÍNDICE DE CONTENIDOS.

RESUMEN	1
PARTE I. FUNDAMENTACIÓN	3
INTRODUCCIÓN	4
1.1. Justificación	4
1.2. MOTIVACIÓN. ESTADO DE ARTE	10
1.3. OBJETIVO GENERAL	15
1.4. HIPOTESIS	16
PARTE II. MARCO TEÓRICO.....	17
CAPÍTULO N°1. Aprendizaje.....	18
1.1. ¿Qué entendemos por aprendizaje?	18
1.2. Estilos de aprendizaje, estrategias de aprendizaje y enfoques de aprendizaje 20	
CAPÍTULO N°2. PROCESOS EDUCATIVOS EN EL NIVEL SUPERIOR DE ENSEÑANZA	26
2.1. Nuevos retos de la Universidad	26
2.2. El tránsito de la Educación Secundaria a la Universidad	28
2.3. Los procesos educativos y estrategias de enseñanza	30
2.4. Rol docente en el proceso de enseñanza	33
2.5. Caracterización de los estudiantes: Generación Z	35
CAPÍTULO N°3. La robótica en los procesos educativos.....	38
3.1 Robótica	38
3.2 La Robótica Pedagógica	39
3.3 El enfoque pedagógico	43
CAPÍTULO 4. PROGRAMACIÓN	46
4.1. Uso versus comprensión	46
4.2. Programación y mercado laboral	48
4.3. La importancia de la enseñanza de la programación en las Escuelas	54
PARTE III. ENCUADRE METODOLÓGICO.....	58
TIPO DE ESTUDIO	59
DISEÑO DE LA INVESTIGACIÓN.	60
UNIVERSO.....	61
Instrumentos de recolección de datos.....	63
PARTE IV. EL CASO DE LA UNNOBA.....	65

CAPÍTULO N°1. ANTECEDENTES.....	67
1.1. La problemática de la deserción y el desgranamiento	67
1.2. Estrategias para mitigar la deserción estudiantil en las carreras del Área de Informática	68
CAPITULO N°2. DESCRIPCIÓN DE LA ASIGNATURA.....	72
3.2 Introducción a la Programación Imperativa	72
3.3 Debilidades en el proceso de enseñanza y aprendizaje de la materia IPI	74
3.4 Estrategias implementadas desde la asignatura Introducción a la Programación Imperativa	76
CAPÍTULO N°3. METODOLOGIA PROPUESTA.....	100
3.5 El Robot Educativo Programable (REP)	100
3.6 Desarrollo de una interfaz para el REP	101
3.7 Justificación de la selección del lenguaje de programación.....	116
CAPÍTULO 4. MARCO REFERENCIAL.....	118
CAPÍTULO N°5. EXPERIMENTO	121
El REP como estrategia para promover habilidades en la Programación Imperativa.	121
5.1 Estilos, estrategias y enfoques de aprendizaje	121
5.2 Ejecución del REP y su interfaz.....	122
CAPÍTULO N°6. DISCUSIÓN FINAL DE LOS RESULTADOS	140
PARTE V. INFORME FINAL	142
Recomendaciones.	147
Futuras líneas de investigación.	148
REFERENCIAS BIBLIOGRÁFICAS	150
ANEXO I Encuesta a estudiantes – año 2017	156
ANEXO II Formulario de Google – año 2018.....	161
ANEXO III Trabajo práctico con ejercicios de REP.....	164

ÍNDICE DE ILUSTRACIONES.

ILUSTRACIÓN 1: OFERTAS POR CATEGORÍA Y POR LOCALIZACIÓN.....	49
ILUSTRACIÓN 2: SUBCATEGORÍA “PROGRAMACIÓN”.....	49
ILUSTRACIÓN 3: OFERTAS LABORALES EN COMPUTRABAJO FILTRADA POR LA PALABRA CLAVE “INFORMÁTICA/TELECOMUNICACIONES”.....	50
ILUSTRACIÓN 4: OFERTAS LABORALES EN COMPUTRABAJO FILTRADA POR LA PALABRA CLAVE “PROGRAMADOR”.....	50
ILUSTRACIÓN 5: CAPTURA DE PANTALLA DEL PORTAL ZONAJOB.....	51
ILUSTRACIÓN 6: FILTRADO POR ÁREA "TECNOLOGÍA".....	52
ILUSTRACIÓN 7: PORTAL BUMERAN FILTRADO POR EL ÁREA "TECNOLOGÍA".....	53
ILUSTRACIÓN 8: CAPTURA DE PANTALLA DEL SITIO “LOGO FOUNDATION”.....	56
ILUSTRACIÓN 9: INSCRIPTOS A LA CURSADA DE IPI EN LA SEDE DE JUNÍN, AÑO 2018.....	62
ILUSTRACIÓN 10: INSCRIPTOS A LA CURSADA DE IPI EN LA SEDE DE PERGAMINO, AÑO 2018 ...	62
ILUSTRACIÓN 11: INSCRIPTOS A LA CURSADA DE IPI EN LA SEDE DE PERGAMINO, AÑO 2019 ...	63
ILUSTRACIÓN 12: INSCRIPTOS A LA CURSADA DE IPI EN LA SEDE DE JUNÍN, AÑO 2018.....	63
ILUSTRACIÓN 13: ÍNDICES DE DESERCIÓN PARA LA CARRERA “LICENCIATURA EN SISTEMAS”. 68	
ILUSTRACIÓN 14: ÍNDICES DE DESERCIÓN PARA LA CARRERA “INGENIERÍA EN INFORMÁTICA” 68	
ILUSTRACIÓN 15: CURSO PARA IPI EN UNNOBA VIRTUAL.....	77
ILUSTRACIÓN 16.....	78
ILUSTRACIÓN 17.....	78
ILUSTRACIÓN 18.....	79
ILUSTRACIÓN 19: CIUDAD POR LA QUE CIRCULA ADA.....	80
ILUSTRACIÓN 20.....	81
ILUSTRACIÓN 21.....	86
ILUSTRACIÓN 22.....	86
ILUSTRACIÓN 23: NIÑOS TRABAJANDO EN EQUIPO EN EL ARMADO DE UN ALGORITMO.	89
ILUSTRACIÓN 24: ALUMNOS DEL CURSO TRABAJANDO AL AIRE LIBRE EN UNO DE LOS EDIFICIOS DE LA UNNOBA.....	89
ILUSTRACIÓN 25: ALGORITMO ARMADO POR UN GRUPO.....	90
ILUSTRACIÓN 26: CAPTURA DE LA WEB DE PILAS BLOQUES.....	92
ILUSTRACIÓN 27: UNO DE LOS DESAFÍOS PARA EL PRIMER CICLO.....	92
ILUSTRACIÓN 28: UNO DE LOS DESAFÍOS PARA EL SEGUNDO CICLO.....	93
ILUSTRACIÓN 29: INTERFAZ DE ALICE AL ABRIR LA APLICACIÓN.	94
ILUSTRACIÓN 30.....	95
ILUSTRACIÓN 31: PROYECTOS COMPARTIDOS CON LA COMUNIDAD.....	96
ILUSTRACIÓN 32: CAPTURA DE PANTALLA DE GOSBTONE AL INICIO.....	97
ILUSTRACIÓN 33.....	98
ILUSTRACIÓN 34: IMAGEN DE REP.....	100
ILUSTRACIÓN 35: IMAGEN DE UNA PLACA ARDUINO NANO.....	103
ILUSTRACIÓN 36.....	104
ILUSTRACIÓN 37: PANTALLA PRINCIPAL DE LA INTERFAZ SEÑALANDO EL BOTÓN “EJECUTAR/ENVIAR”.....	105
ILUSTRACIÓN 38: CAPTURA DE PANTALLA DE LA INTERFAZ.....	106
ILUSTRACIÓN 39: ESQUELETO DE LA INTERFAZ, DIVIDIDO EN SEIS PANELES.....	106
ILUSTRACIÓN 40.....	107
ILUSTRACIÓN 41.....	107

ILUSTRACIÓN 42: REPRESENTACIÓN GRÁFICA DE UN EJEMPLO.	108
ILUSTRACIÓN 43: VENTANA DE UNA FUNCIÓN VACÍA.	113
ILUSTRACIÓN 44: VENTANA DE UNA FUNCIÓN EN ESTADO DE “CONSTRUCCIÓN”.	114
ILUSTRACIÓN 45.	115
ILUSTRACIÓN 46.	116
ILUSTRACIÓN 47: PREGUNTA DOS.	123
ILUSTRACIÓN 48: RESUMEN DE RESPUESTAS DE LA PREGUNTA DOS.	123
ILUSTRACIÓN 49: PREGUNTA TRES.	123
ILUSTRACIÓN 50: RESUMEN DE RESPUESTAS DE LA PREGUNTA TRES.	124
ILUSTRACIÓN 51: PREGUNTA CUATRO.	124
ILUSTRACIÓN 52: RESUMEN DE RESPUESTAS DE LA PREGUNTA CUATRO.	125
ILUSTRACIÓN 53: PREGUNTA SEIS.	125
ILUSTRACIÓN 54: RESUMEN DE RESPUESTAS DE LA PREGUNTA SEIS.	126
ILUSTRACIÓN 55: PREGUNTAS NUEVE Y DIEZ.	126
ILUSTRACIÓN 56: RESUMEN DE RESPUESTAS DE LA PREGUNTA NUEVE.	127
ILUSTRACIÓN 57: RESUMEN DE RESPUESTAS DE LA PREGUNTA DIEZ.	127
ILUSTRACIÓN 58.	134

ÍNDICE DE TABLAS.

TABLA 1: DATOS DE LA ASIGNATURA “INTRODUCCIÓN A LA PROGRAMACIÓN IMPERATIVA” EN LA UNNOBA.....	11
TABLA 2: DATOS DE LA ASIGNATURA “INTRODUCCIÓN A LA PROGRAMACIÓN IMPERATIVA” EN LA UNNOBA.....	75
TABLA 3: CANTIDAD DE INSCRIPTOS DE LOS ÚLTIMOS CINCO (5) AÑOS EN IPI.	120
TABLA 4: COMPARACIÓN DE RESULTADOS FINALES DE LA ASIGNATURA, TOMANDO PARA EL AÑO 2019 SÓLO ESTUDIANTES DE LA SEDE JUNÍN.....	133
TABLA 5: COMPARACIÓN DE RESULTADOS FINALES DE LA ASIGNATURA, TOMANDO PARA CADA AÑO ESTUDIANTES DE AMBAS SEDES	133

RESUMEN

La presente investigación, titulada “*Aprendiendo a programar con el Robot Educativo Programable (REP)*” se concreta con el objetivo de delinear una propuesta pedagógica y metodológica que contribuya al aprendizaje del Paradigma Imperativo de la programación de computadoras en los estudiantes de primer año de las carreras del área de Informática de la Universidad del Noroeste de la Provincia de Buenos Aires (UNNOBA), utilizando como recurso didáctico el REP.

Si bien el dominio de los fundamentos de programación de computadoras es una habilidad esencial a ser desarrollada por los estudiantes de las carreras de la mencionada área, los datos estadísticos sobre los resultados de las asignaturas afines y los informes y tutorías de los ingresantes, revelan una acusada dificultad en la introducción de los estudiantes a la programación imperativa, reflejado en el bajo rendimiento académico, un elevado índice de deserción y un significativo desgranamiento en los años posteriores.

Por tal motivo se indagará sobre la aplicación de robots como herramienta educativa en nivel superior, se determinará el modo en que debe utilizarse un robot para realizar actividades educativas dentro del paradigma imperativo de la programación de computadoras, se diseñará una interfaz para su utilización, se propiciará la experimentación y se estimularán las competencias asociadas a la resolución de problemas y se facilitará el aprendizaje de un lenguaje de programación a través del REP.

El estudio se desarrollará mediante un diseño experimental puro y el producto final será un documento que incluya estrategias áulicas que permitan incorporar instrumentos tecnológicos no tradicionales, como el Robot Educativo Programable, a

las actuales propuestas educativas de la UNNOBA en lo referido a la programación imperativa, las cuales, aplicadas de manera sistemática puedan a mediano plazo también dar respuesta a la problemática de la deserción y el desgranamiento estudiantil.

Se realizará un estudio de caso en el marco de la materia “*Introducción a la Programación Imperativa*” (IPI), perteneciente al primer año de estudios de las carreras del área de Informática de la mencionada universidad.

Palabras claves: *Robots - Herramientas educativas - Programación - Estrategias didácticas- Programación Imperativa.*

PARTE I. FUNDAMENTACIÓN

INTRODUCCIÓN

1.1. Justificación

Las tecnologías emergentes, como su nombre lo indica, son innovaciones en desarrollo que en un futuro no muy lejano cambiarán la forma de vivir del ser humano, brindándole una mayor facilidad a la hora de realizar sus actividades. Estas se presentan cada vez más como una necesidad en el contexto de una sociedad donde los cambios acelerados, el aumento de la demanda y actualización de la información y de nuevos conocimientos se convierten en una exigencia permanente.

De este modo, las tecnologías no solo impactan en el modelo de sociedad, en las costumbres de los pueblos y en las diversas formas de relacionarse y comunicarse, sino también, y de forma cada vez más evidente en el ámbito educativo, y como correlato en el proceso de enseñanza y aprendizaje.

En relación a esto último, el sistema educativo no debe permanecer ajeno a los desafíos que el actual contexto tecnológico le impone, debiendo reestructurar y reorientar sus propuestas pedagógicas a la luz de estos cambios, para formar a las nuevas generaciones en habilidades de pensamiento como complemento de los contenidos específicos que fueron el eje de la enseñanza hasta el momento.

La brecha entre lo que se espera de los nuevos profesionales y cómo ellos se gradúan de la Universidad podría acotarse si se aúnan los esfuerzos para formar a los jóvenes en prácticas de pensamiento desde el comienzo de su formación. En este sentido, podría considerarse que llevar a cabo estas acciones desde el primer año de toda carrera universitaria, incluso desde la educación secundaria, también contribuiría a la adaptación y permanencia del alumnado, favoreciendo su desempeño académico durante todo su proceso formativo.

El mundo tecnológico se encuentra en constante desarrollo y ello promete que quienes adquieran habilidades y conocimientos en esta área pueden llegar a tener más posibilidades de crecimiento profesional que en otras carreras. En lo que respecta puntualmente al área de las Ciencias de la Computación, la demanda laboral actual se orienta a la búsqueda de recursos humanos que puedan realizar tareas en las que se involucre la creatividad, es decir la capacidad para “dar” nacimiento a una idea o innovar una ya existente.

De este modo, la Universidad como parte del sistema educativo debería, entre otras cuestiones, capacitar a sus futuros graduados en una formación centrada en prácticas de resolución de problemas, que incentive la creatividad para poder alcanzar un desempeño satisfactorio en el campo profesional (Rosas, Zúñiga, Fernández y Guerrero, 2017).

Cuando una persona programa lo que hace básicamente es diseñar una solución para un problema a través de algoritmos, y como no existe forma de determinar una única solución tampoco existe manera de establecer que un algoritmo sea el único que soluciona tal problema. De este modo, esta “libertad” indefectiblemente pone en juego la creatividad de la persona, en este sentido está dada la utilidad de la programación.

En lo que respecta al caso puntual de la UNNOBA, se advierte una gran dificultad por parte de los estudiantes de las carreras del área de informática, en la introducción a la programación imperativa, reflejado en el bajo rendimiento académico, en el elevado índice de deserción y el significativo desgranamiento en los años posteriores. De acuerdo a los planes de estudio en la UNNOBA, la asignatura “Programación Imperativa” (PI) es correlativa a IPI. Durante el segundo año es “Introducción a Objetos” la asignatura correlativa a PI. Sucede históricamente en esta universidad que

efectivamente llega un número considerablemente menor de los estudiantes que iniciaron en IPI a “Introducción a Objetos”.

En este sentido será necesario realizarse ciertos cuestionamientos tales como: ¿Qué dificultad especial subyace en el aprendizaje de la programación? ¿Estará relacionado con lo abstracto de la tarea en sí? ¿Habrá correlación con el alto nivel de simbología y razonamiento utilizado? ¿Cómo se accede al aprendizaje actualmente? ¿Hasta qué punto debe utilizarse la memoria? ¿En qué medida se pone en juego el razonamiento?

Anteriormente, la enseñanza de la programación de computadoras se centraba casi exclusivamente en el uso de la memoria, sin embargo hace un tiempo se viene haciendo hincapié en la importancia del uso del razonamiento. Entonces, puede ser que la clave esté en confluir entre memoria y razonamiento, debiendo utilizarse ambas habilidades de modo tal que el aprendiz pueda construir su propio conocimiento. En relación a lo anterior, era usual abordar ejercicios para “aprender” y luego se aplicaba mecánicamente lo “aprendido” en otros ejercicios similares. Pero ¿acaso no será más redituable en términos educativos brindar al estudiante un entorno de exploración y análisis para que pueda poner en práctica lo que conoce y lo que va conociendo durante el proceso de aprendizaje? Probablemente, trabajando mecanismos que lo lleven a descubrir los conceptos de un modo más constructivo, se puedan obtener mejores resultados.

Intentado responder estos interrogantes, y apoyada en la experiencia y trayectoria como docente universitaria desde el año 2005 en asignaturas relacionadas con la enseñanza de la programación de computadoras, se han llevado a cabo numerosos esfuerzos e introducido múltiples cambios para responder a estas cuestiones. Sin embargo los resultados no han sido los esperados.

Teniendo en cuenta que aún no se ha probado en la UNNOBA el uso de la robótica para la enseñanza de la programación, se propone en el marco de la presente tesis, un estudio exploratorio que brinde un contexto de aplicación adecuado al uso de una herramienta pedagógica poco estudiada como es el caso del REP. Este proyecto tiene como principal aportación la de lograr en los alumnos el fortalecimiento en la comprensión de las ciencias, utilizando la robótica educativa como herramienta pedagógica y principal motor para su motivación.

Es importante destacar que a los efectos de la presente investigación, no se concibe a la robótica como un fin en sí mismo, sino como un medio para alcanzar ciertos aprendizajes significativos.

Se plantea entonces a la robótica pedagógica como una alternativa didáctica que, a la par de los métodos ya establecidos, propende por nuevos enfoques que promuevan en los estudiantes intereses que coadyuven en la creación de ambientes para el aprendizaje en el que se encuentren circunstancias favorables para la construcción de conceptos y de su interpretación personal de la realidad.

Se parte de la premisa que un ambiente de aprendizaje adecuado para actividades didácticas con robots puede ayudar al aprendizaje de contenidos teóricos, por lo cual el uso educacional de la robótica provee a los estudiantes un ambiente de aprendizaje empírico, con un notable potencial para el apoyo del aprendizaje activo (Monsalves, 2011).

Todo esto se realizará mediante secuencias didácticas y entornos de aprendizaje donde el alumno tendrá la necesidad de empezar a construir nuevos conocimientos. No olvidemos que uno de los objetivos de la Educación Superior es formar estudiantes autónomos que sean capaces de dirigir su propia experiencia para generar aprendizaje. (Padilla, 2003).

El mencionado estudio se desarrollará mediante un diseño experimental puro y el producto final será un documento que defina estrategias que permitan incorporar instrumentos tecnológicos no tradicionales, como el Robot Educativo Programable, a las actuales propuestas educativas de la UNNOBA en lo referido a la programación imperativa. En ese sentido está dada su utilidad ya que viene a sentar precedente acerca de una herramienta que aun en esta universidad no se ha trabajado.

La propuesta se implementará bajo un enfoque pedagógico que tenga en cuenta el ambiente de aprendizaje, los estilos de aprendizaje, la planeación de las actividades, los recursos y la metodología a desarrollar. Estas características del modelo pedagógico garantizarán la construcción y reconstrucción del conocimiento por parte del estudiante (López Ramírez, 2013). En el marco de la presente tesis, la meta es utilizar la robótica para fomentar el aprendizaje por indagación, orientado a resolver problemas.

En cuanto al impacto, vale reiterar que se espera que esta investigación enriquezca las propuestas educativas para la enseñanza de la programación imperativa en la UNNOBA. Con tal fin se elaborarán estrategias que utilicen al REP como herramienta y luego se las analizarán evaluando la posibilidad de su aplicación en la asignatura “Introducción a la Programación Imperativa” (IPI) correspondiente al primer año de las carreras informáticas de la UNNOBA.

En el marco de esta tesis se trabajará con conceptos tales como diseño de interfaz, didáctica de la programación y herramientas educativas actuales. La investigación abordará los siguientes ejes temáticos: uso de robots como herramienta educativa, tipos y niveles de interfaces y su uso en el plano educativo, conceptos y habilidades para el desarrollo de interfaz de robots, utilización de estrategias en propuestas educativas para promover habilidades en programación imperativa, estrategias para motivar el aprendizaje de los estudiantes.

La efectividad de las estrategias educativas se verá reflejado no solo en la calificación de los estudiantes y la promoción de la asignatura, sino también, y en términos cualitativos, en un desempeño integral del alumno en términos académicos y de cara al crecimiento profesional.

En lo que concierne a la UNNOBA, ésta se verá beneficiada en tanto y en cuanto los estudiantes de las carreras afines a la programación de computadoras logren un número más elevado de regularizados, y en definitiva se obtendrán graduados con mayor calidad de preparación profesional. No olvidemos que el nivel de retención de estudiantes de una universidad es un indicador para la medición de su calidad (Muñoz, Noel, Barría, Barcelos y Silveira; 2015).

En cuanto a la viabilidad y factibilidad de la propuesta, el estudio se considera viable ya que quien lo conduce es parte del equipo docente de la asignatura sobre la que se realizará el estudio de caso, y por otra parte es investigadora integrante del Instituto de Investigación y Transferencia de Tecnológica (ITT). Además, cabe destacar que la tesista coordina actividades en escuelas primarias y secundarias dirigidas a incentivar carreras vinculadas a las TIC en el marco del programa “PROGRAM.AR”. Esta iniciativa fue creada en el año 2013 por la FUNDACIÓN DR. MANUEL SADOSKY¹ y su objetivo es impulsar la enseñanza y el aprendizaje de las Ciencias de la Computación en la Escuela Argentina. En este sentido, la viabilidad técnica está dada en un doble sentido, por un lado la coherencia interna de la propuesta y por el otro la idoneidad profesional de quien lo conduce.

A su vez es factible ya que se cuenta con la participación de los recursos humanos directamente implicados (estudiantes de primer año de las carreras del área de

¹ Para mayor información acceder al sitio <http://www.fundacionsadosky.org.ar/presentacion-institucional/>

informática). Estrechamente vinculado a lo anterior, se puede destacar la viabilidad política de la presente tesis, garantizada por el hecho que desde el año 2005 su autora se desempeña como docente de la Universidad en las asignaturas “Introducción a la Programación Imperativa” y “Programación Imperativa” y desde el año 2010 en el rol de tutora de los primeros años de las carreras Ingeniería en Informática, Licenciatura en Sistemas y Analista de Sistemas. Motivo por el cual no se encontraran resistencias por parte de la institución para desarrollar allí la propuesta.

En relación al tiempo previsto para su ejecución (período 2018-2019) se considera suficiente para ejecutar la intervención y evaluar sus resultados. Finalmente con respecto a la viabilidad de finalización, se estima que el cuerpo docente de la UNNOBA podrá valerse de lo investigado y adaptar sus propuestas educativas haciendo uso de estas estrategias u otras similares.

1.2. MOTIVACIÓN.

El impulso que motiva esta propuesta de tesis es el mensaje del documento audiovisual “*What most schools don't teach*” (Lo que la mayoría de las escuelas no enseñan) publicado por Code.org. Este comienza con la siguiente sentencia de Steve Jobs²: “*Todo el mundo debería aprender a programar porque te enseña a pensar*”. Sin embargo, como se explicó anteriormente, si se analizan los datos estadísticos sobre los resultados de las asignaturas afines a la programación de computadoras en la UNNOBA y se tienen en cuenta informes y tutorías de los ingresantes a las carreras del área de informática, se advierte una acusada dificultad en la introducción de los estudiantes a la programación imperativa.

² Steve Jobs fundó en 1975 Apple Computer Inc. junto a Steve Wozniak. Esta empresa comenzó armando ordenadores en forma artesanal y para 1983 ya había entrado en el puesto 411 de Fortune 500, convirtiendo su ascenso en el más rápido de la historia empresarial. Fuente: <http://www.muyinteresante.es/tecnologia/articulo/6-curiosidades-sobre-la-vida-de-steve-jobs>.

A continuación se incluye un cuadro con los datos de asistencia y regularización en la asignatura Introducción a la Programación Imperativa (IPI) dictada en la UNNOBA entre los años 2010 y 2016³:

Año	Asistentes	Regularizados
2016	42,55%	20,57%
2015	54,35%	18,12%
2014	54,41%	30,15%
2013	52,76%	24,84% ⁴
2012	52,76%	29,15%
2011	67,00%	33,00%
2010	60,00%	28,00%

Tabla 1: Datos de la asignatura “Introducción a la Programación Imperativa” en la UNNOBA.

En la columna “Asistentes” de la Tabla 1 puede observarse el porcentaje de estudiantes que efectivamente asistieron a clases, para lo que se registra la asistencia a clase. Por otra parte en la columna “Regularizados” se discriminan aquellos que han aprobado la cursada de la asignatura, Tras un breve análisis de los datos se puede observar que el porcentaje de los asistentes a clase ha disminuido año tras año. Particularmente, entre 2010 y 2012 los estudiantes regularizados oscilan entre el 28 y el 33 por ciento.

Con el objetivo de incrementar el porcentaje de estudiantes regularizados en la asignatura, en el 2013 se decidió cambiar lenguaje de programación utilizado para abordar los conceptos de la asignatura. Además desde esa fecha se han aplicado algunas estrategias que involucraron la reconfiguración de las dinámicas de las clases prácticas y teóricas, la capacitación del plantel docente en didáctica de la programación

³ En la Tabla 1 se destaca la fila correspondiente al 2013 pues fue el año en el que se cambió el lenguaje de programación utilizado como herramienta para abordar los conceptos de la asignatura.

imperativa y la organización de encuentros presenciales y virtuales para facilitar el intercambio dentro del equipo. Si bien los esfuerzos y cambios introducidos generaron una mejoría en los resultados, no se ha alcanzado el porcentaje de “Regularizados” deseado por la institución.

Por tanto la experiencia y trayectoria de la autora de la presente Tesis, como docente universitaria desde el año 2005 en asignaturas relacionadas con la enseñanza de la programación de computadoras, suscitó la búsqueda de nuevas estrategias para la enseñanza de la programación de computadoras. Además, su rol de tutora desde el año 2010 en los primeros años de las carreras Ingeniería en Informática, Licenciatura en Sistemas y Analista de Sistemas le ha proporcionado una mirada holística respecto a las dificultades que se pueden presentar en estudiantes del área de informática.

Este contexto hace evidente la importancia de una inminente intervención a través de una acción educativa innovadora por lo cual se considera útil y conveniente probar en la UNNOBA el uso de la robótica para la enseñanza de la programación.

Al respecto, cabe destacar que desde el Instituto de Investigación y Transferencia de Tecnológica, un grupo de investigadores han trabajado en la construcción de un prototipo del Robot Educativo Programable (REP). Esta labor se llevó a cabo a fin de contar con una herramienta propia de la universidad para trabajar a futuro en la enseñanza de la programación de computadoras. La construcción del prototipo se ha desarrollado en el marco de un trabajo de investigación denominado “Herramientas didácticas y nuevas formas de interacción para la inclusión de TICs en la enseñanza” cuyo objetivo general es analizar y desarrollar herramientas y estrategias didáctico-pedagógicas que permitan el diseño y la evaluación de actividades para la inclusión de las TIC en las actividades áulicas de las escuelas primarias y secundarias en articulación con el nivel superior.

Este estudio propone entonces el diseño de una interfaz para el Robot Educativo Programable a fin de integrarlo y utilizarlo como herramienta para el caso de estudio seleccionado.

Por todo lo expuesto se plantea este trabajo de tesis como un intento de delinear una propuesta pedagógica y metodológica para el aprendizaje del Paradigma Imperativo de la programación de computadoras en las carreras del área de Informática de la UNNOBA, utilizando como recurso didáctico el REP. Se estudiará puntualmente el caso de la asignatura Introducción a la Programación Imperativa, común al primer año de estudios de estas carreras.

La revisión de la literatura de la última década, revela que la mayoría de los estudios han sido realizados en otros países, y aunque algunos se han ocupado de variables similares, no han perseguido los mismos objetivos. Este es el caso del estudio de Román- Graván, Hervás -Gómez y Guisado-Lizar (2017) que refleja los resultados de experiencias realizadas con el uso de diferentes robots que se están comercializando en España y en el que han participado estudiantes de la asignatura de TIC del Grado de Educación Infantil y Primaria de la Facultad de Ciencias de la Educación de la Universidad de Sevilla. Los resultados del estudio revelan que los alumnos han estado muy motivados y han reaccionado muy positivamente ante la introducción de la robótica en el currículum académico.

Otro antecedente que se considera importante mencionar, es el de Pinto Salamanca, Barrera Lombana y Pérez Holguín (2010) en el cual se describe la implementación de un robot móvil de configuración diferencial, construido con el set de piezas del kit de robótica Lego Mindtorms NXT, como apoyo a los procesos de enseñanza y aprendizaje en los niveles de educación preescolar y educación básica primaria, en tres instituciones educativas del departamento de Boyacá, Colombia. Los

resultados de su aplicación indican, de forma general, que se hace evidente el potencial de los ambientes para el aprendizaje desarrollados a partir de la robótica educativa, como nueva propuesta didáctica que responde a los requerimientos de sociedades contemporáneas que reclaman nuevos métodos de enseñanza. Sus resultados también evidencian que el desarrollo de estas prácticas debe estar guiado por personal con formación en didáctica y pedagogía, que aporte su conocimiento y experiencia en el ámbito educativo. Si bien los docentes muestran interés e iniciativa en el complemento de sus clases con herramientas que hagan uso de la tecnología, manifiestan desconocimiento en el manejo de herramientas didácticas que usen tecnologías recientes.

En lo que respecta puntualmente a nuestro país, no se dispone de abundante información que aporte evidencia empírica acerca de la temática en cuestión. Casi la totalidad de las experiencias, tal como se indicó con anterioridad, fueron desarrolladas en los Niveles Primario y Secundario de enseñanza. En este sentido, un antecedente que resulta relevante mencionar es el de Banchoff Tzancoff, Martín, Gómez y López (s.f) titulado “Experiencias en robótica educativa. Diez años trabajando con escuelas argentinas”. Este trabajo describe las experiencias realizadas con Robots y otros recursos similares en la última década, a partir de la realización de un análisis de las distintas intervenciones. Los resultados de las experiencias indican que, para el abordaje de contenidos relacionados con tecnologías, se debe fomentar el trabajo en equipo, la producción de conocimiento y el compromiso social, en el sentido de una ciudadanía digital. Por otro lado, el estudio recomienda trabajar con la utilización de un grupo de sensores dado que potencia la diversidad de actividades a realizar, permitiendo a los estudiantes ser partícipes de la resolución y desarrollo del programa para la resolución de un desafío. El estudio concluye señalando que, si bien la utilización de robots en el

ámbito educativo en forma masiva ha comenzado hace poco tiempo, los resultados incipientes reflejan motivación de entusiasmo, interés por temáticas relacionadas al conocimiento científico-tecnológico y fortalecimiento de las habilidades relacionadas con el pensamiento computacional.

Respecto al Nivel Superior Universitario, no se han podido localizar en la literatura afín trabajos orientados a definir estrategias educativas con respecto al aprendizaje de la programación de computadoras utilizando la robótica como herramienta pedagógica para fomentar el aprendizaje por indagación.

Tomando estos antecedentes se plantea como objetivo general de este trabajo:

1.3. OBJETIVO GENERAL

Establecer estrategias educativas con respecto al paradigma imperativo de la programación de computadoras utilizando como herramienta al Robot Educativo Programable (REP), en el primer año de estudios de las carreras del área de informática de la UNNOBA, durante el periodo 2018-2019.

1.3.1. Objetivos específicos

- Estudiar las estrategias actualmente utilizadas para la enseñanza de la programación imperativa en la Universidad Nacional del Noroeste de la Provincia de Buenos Aires.
- Indagar sobre la aplicación de robots como herramienta educativa en el Nivel Superior.
- Determinar de qué modo debe utilizarse un robot para realizar actividades educativas dentro del paradigma imperativo de la programación de computadoras.

- Establecer de qué manera se podría motivar a los estudiantes mediante el uso del Robot Educativo Programable.
- Diseñar una interfaz para el uso del Robot Educativo Programable.

1.4. HIPOTESIS

- Se estima que el uso de robots podría tener un alto potencial de permitir la aplicación más tangible y concreta de las habilidades de programación.
- La interfaz para el uso del REP permitirá establecer una forma diferente de acercar a los estudiantes a los contenidos, y motivarlos mediante el uso de un robot, siempre y cuando, a través de la intervención del docente, se logre un diseño de actividades motivadoras y estimulantes.

PARTE II. MARCO TEÓRICO

CAPÍTULO N° 1. APRENDIZAJE

Cuando se habla de procesos o actos educativos no se puede escapar del término aprendizaje. Parafraseando a Manes (2017) el aprendizaje es tan importante y tan central en la vida que por eso se vuelve primordial tratar de comprender qué es y cómo se produce. Ambas cuestiones intentarán responderse en el presente capítulo para poder luego reflexionar acerca de cómo pueden mejorarse los procesos educativos. Será importante iniciarse en estos significados para luego poder adentrarse en el aprendizaje de la programación de computadoras.

Tal como se presentará a continuación, el proceso de maduración cognoscitivo se encuentra asociado a una modalidad particular de cada persona, es decir una manera personal de acercarse al conocimiento. La modalidad de aprendizaje opera como una matriz que está en permanente reconstrucción y sobre la cual se van incluyendo los nuevos aprendizajes que van transformándola (Fernández, 2002, 2008 y 2010). Lo expuesto lleva a reconocer la existencia de diferencias individuales entre los estudiantes que determinan modos distintos de aprender. De ahí, la importancia de desplegar estrategias de enseñanza sobre la base del diagnóstico del procesamiento de información de los estudiantes (Sánchez, 2002).

1.1. ¿Qué entendemos por aprendizaje?

Aprender es apropiarse, apropiación que se da a partir de una elaboración objetivante y subjetivante. La elaboración objetivante permite apropiarse del objeto seriando y clasificando. En cambio, la elaboración subjetivante tratará de reconocer, de apropiarse a partir de la experiencia única e intransmisible que haya tenido el sujeto con el objeto de conocimiento.

El aprendizaje ocurre durante toda la vida. A diario se procesa información a partir de la cual se construyen esquemas mentales, lo que permite luego reflexionar, decidir y actuar en consecuencia. Por esquemas mentales se entiende al proceso cognitivo o conceptual que ayuda a organizar e interpretar cierta información (Vergara, 2017).

La cognición es por tanto, la cualidad de la mente humana para captar y analizar la realidad que nos rodea, por medio de la percepción y de los órganos del cerebro (García Bartolomé, 2018). Dicho en otros términos, es la capacidad de asimilar y procesar datos, valorando y sistematizando la información a la que se accede a partir de la experiencia y la percepción.

El aprendizaje le permite al sujeto transitar por los diferentes esquemas mentales, los cuales se van organizando de distinta manera a medida que se reciben diferentes estímulos en la interacción con el entorno. En este sentido el aprendizaje es relacional, lo que significa que cada persona, incluso sin advertirlo, incorporará habilidades y conocimientos a partir de lo que ya sabe, de modo que la relación con lo anterior es inevitable (Regader, 2015).

A través las nuevas experiencias, los esquemas existentes se ajustan o se modifican, proceso conocido como asimilación y acomodación. En el primero, se incorpora nueva información a los esquemas preexistentes, mientras que en la acomodación, estos se modifican o bien, pueden formarse nuevos esquemas, a partir de las nuevas experiencias.

1.2. Estilos de aprendizaje, estrategias de aprendizaje y enfoques de aprendizaje

De lo expuesto anteriormente, se puede deducir que el aprendizaje humano es un fenómeno complejo, sobre todo si se tiene en cuenta que en cada persona se llevan a cabo múltiples procesos cognoscitivos al momento de aprehender un conocimiento. Los procesos cognoscitivos refieren al conjunto de técnicas, inventivas, iniciativas intelectuales que enriquecen los esquemas mentales antes mencionados (Ventura, Gagliardi y Moscoloni, 2012).

Diferentes paradigmas teóricos, que durante el siglo XX se han enfocado en explicar la manera en que ocurre el aprendizaje humano, coinciden en que el proceso de maduración cognoscitivo se encuentra asociado a una modalidad peculiar y particular de cada persona. Se trata pues de una manera, una forma de comportamiento intelectual dinámica, que se perfecciona durante procesos epistemológicos (Ventura et.al., 2012).

Existen entonces diferencias individuales entre los estudiantes que determinan modos distintos de aprender. En palabras de Cano García (2000) tales diferencias pueden analizarse a través de distintos constructos entre los que se destacan los *estilos*, las *estrategias* y los *enfoques de aprendizaje* (en Freiberg Hoffmann, Berenguer, Fernández Liporace y Ledesma, 2017).

1.2.1. Los estilos de aprendizaje

El estudio sobre los estilos de aprendizaje se inicia durante la década del 70 desde una perspectiva psicológica cognitiva. A partir de este enfoque, se proponía implementar nuevas estrategias de enseñanza sobre la base del diagnóstico del procesamiento de información de los estudiantes (Sánchez, 2002).

Este paradigma cuestiona una tendencia relativamente generalizada, basada en la implementación de ciertas condiciones, métodos y estrategias de enseñanza de manera invariante, para promover aprendizajes significativos en los estudiantes (Espinosa Lerma, 2008). El uso de estos métodos se aplicaba bajo el supuesto de que los alumnos llevan a cabo los mismos procesos cognitivos de adquisición y elaboración del conocimiento, especialmente en contextos formales de enseñanza y aprendizaje.

En contraposición a esta postura, se demuestra la existencia de marcadas diferencias individuales en los procedimientos y mecanismos cognitivos básicos implicados en el aprendizaje. Estas diferencias reciben el nombre de *estilos de aprendizaje* y aluden a configuraciones de rasgos relativamente estables en torno a los modos típicos en que cada estudiante prefiere percibir, procesar y comprender la información en un contexto de enseñanza y aprendizaje (Felder y Brent, 2005; Alonso García y Gallego, 2010). Es decir, remiten a “*cómo perciben la información, de qué manera la procesan, cómo forman conceptos, reaccionan y se comportan en situaciones cotidianas de aprendizaje*” (Hervás y Hernández, 2006, p. 2).

En definitiva, de acuerdo a este paradigma, cada ser humano responde a una forma particular y predominante de aprendizaje, que le permite interrelacionarse con su ambiente y que implica aspectos cognitivos referentes a su personalidad (características fisiológicas y psicológicas que los diferencian unos de otros). Es decir, aunque tengan las mismas motivaciones, el mismo nivel de instrucción, la misma edad o estén estudiando el mismo tema, cada persona aprende con su propio estilo, a su propio ritmo y con sus propias estrategias, incluso con mayor o menor eficacia (Cazau, 2004).

El desarrollo del aprendizaje se establece de acuerdo a las características de las personas que procesan, internalizan y utilizan la información para habilidades académicas (Aparicio, 2004); esto implica hacer uso de sus facultades cognitivas de

acuerdo al proceso de procesamiento de la información y, tal como expresa Padilla (2006), representa la forma típica en que el sujeto percibe, piensa solucionar problemas y se relaciona con otros factores biopsicosociales particulares del individuo (en Juárez Landín, 2016).

A modo ilustrativo, se retoman los principales aportes relacionados con los estilos de aprendizaje desarrollados durante el siglo XX. Encontramos entre ellos, la propuesta de Jung (1921), quien presentó cuatro categorías de estilos de aprendizaje: los que se centran en las *sensaciones*, los *intuitivos*, los que utilizan el *pensamiento* y los *sentimentales*. Posteriormente, a mediados de siglo, surgen otras propuestas; por ejemplo Witkins (1972) identificó dos estilos cognitivos, los *dependientes de campo* y los *independientes de campo*; y Kolb (1981) encontró cuatro diferentes estilos de aprendizaje: los *divergentes*, los *asimiladores*, los *convergentes* y los *acomodadores*. Por su parte, Felder y Silverman (en Figueroa et.al., s.f) presentan un nuevo enfoque, clasificando a los estudiantes según su forma de aprender en los siguientes binomios: - *sensitivos* (concretos, prácticos, orientados hacia los hechos y los procedimientos) o *intuitivos* (conceptuales, innovadores, orientados hacia las teorías), - *visuales* (prefieren la presentación visual del material tal como películas, cuadros, o diagramas de flujo) o *verbales* (prefieren las explicaciones escritas o habladas), - *inductivos* (prefieren la información que deviene desde lo específico hacia lo general) o *deductivos* (prefieren la información que deviene desde lo general hacia lo específico), - *activos* (aprenden manipulando las cosas y trabajando con otros) o *reflexivos* (aprenden pensando acerca de las cosas y trabajando solos), y - *secuenciales* (aprenden poco a poco en forma ordenada) o *globales* (aprenden de forma holística).

Entre los modelos señalados se inserta el propuesto por Alonso, Gallegos y Honey (1997), adaptado de Honey y Mumford (1986), con base en el trabajo de Kolb (1981).

La propuesta se resume en cuatro estilos, a saber: 1. *Activos*; personas creativas, aventureras y entusiastas ante las nuevas tareas y experiencias. Se caracterizan por perseguir los desafíos y por aburrirse con las actividades a largo plazo. Son generadores de ideas y solucionadores de problemas. 2. *Reflexivos*, son personas que consideran las experiencias que pueden ser observables desde diferentes puntos de vista. Analizan los datos muy exhaustivamente considerando todas las alternativas posibles antes de tomar alguna decisión o llegar a alguna conclusión. Se caracterizan por ser receptivos, analíticos, exhaustivos, observadores, detallistas, investigadores y prudentes. 3. *Teóricos*, se basan en teorías lógicas y complejas que integran de acuerdo a lo observado. Se caracterizan por ser perfeccionistas y por profundizar en sus sistemas de pensamiento con la finalidad de determinar principios, teorías y modelos. Huyen de lo subjetivo y de lo ambiguo, son metódicos, lógicos, objetivos, críticos, sistemáticos, pensadores, buscadores de teorías, buscadores de conceptos y exploradores. 4. *Pragmáticos*, son personas que aplican de manera práctica las ideas y experimentan con ellas en la primera oportunidad que tengan. Son muy impacientes y tienen como premisa que siempre se puede hacer mejor y si funciona y responde a la necesidad, es bueno. Son prácticos, experimentadores, eficaces, directos, realistas, rápidos, concretos, seguros de sí, solucionadores de problemas y planificadores de acciones.

Esta última categoría permite trazar el estilo de aprendizaje de los estudiantes participantes en el presente estudio. No obstante, parafraseando a Felder (2004), es importante tener en cuenta que un mismo estudiante puede presentar varias características por lo que el docente debería ser capaz de adaptar su estilo de enseñanza a los estilos de aprendizaje, de tal forma de no afectar negativamente su rendimiento o su actitud frente a los contenidos (en Muñoz, Noël, Barría, Barcelos y Silveira, 2015). Igualmente, los estilos de aprendizaje no necesariamente permanecen invariables a lo

largo del tiempo. Esto significa que, dependiendo del ambiente o contexto de aprendizaje en el que se desenvuelva el estudiante, éste podría o no modificar ciertas características cognitivas que le permitan adaptarse a la realidad educativa en la que se encuentre (Freiberg Hoffmann et.al., 2017).

1.2.2. Estrategias de aprendizaje

Las estrategias de aprendizaje se definen como la metodología empleada por el estudiante en la resolución de tareas específicas a la hora de aprender (Ventura et.al., 2012). Es decir, se refieren al conjunto de actividades y técnicas que se despliegan durante la situación de aprendizaje, para poder influir sobre los procesos cognoscitivos y afectivos, y alcanzar un determinado fin.

Se diferencian tres tipos principales de estrategias: *cognitivas* (remiten a técnicas para aprender, codificar, comprender y recordar la información en función de determinadas metas de aprendizaje), las *metacognitivas* (se vinculan con la planificación, control y evaluación que hacen los alumnos sobre sus propios procesos mentales) y las *afectivas* (se relacionan con los motivos, intenciones, metas y emociones que intervienen en la asimilación de conocimientos) (Escurra Mayaute, 2006; Valle, González, Cuevas, y Fernández, 1998).

En el ámbito universitario, las estrategias de aprendizaje juegan un papel fundamental. Su carácter intencional permite trazar un plan de acción de modo tal que, a partir de la información que se desea incorporar, sea posible seleccionar aquellas conductas que faciliten la tarea. Así, se vuelve posible introducir mejoras en la tarea académica a través de tres intervenciones educativas que son definidas por Beltrán (2003) como: *prevención* (identificación y reemplazo de estrategias poco eficaces por otras de mayor eficacia), *optimización* (estimulación del empleo de estrategias eficaces

que el estudiante pone en juego en el presente) y *recuperación* (localización y mejora de estrategias asociadas al bajo rendimiento) (en Freiberg Hoffmann et.al., 2017).

1.2.3. Enfoques de aprendizaje

Por su parte, los enfoques de aprendizaje se definen como los procesos cognitivos que posibilitan a los estudiantes percibir y procesar la información proveniente de las diferentes actividades académicas (Biggs, 1988). Ventura et.al., (2012) señalan dos formas diferentes de aprender: una refiere a un tipo de acercamiento superficial con la tarea, donde la información se asimila textual y memorísticamente, y la otra alude a una modalidad profunda, que intenta otorgar significados a través de la identificación de las ideas principales del material aprehendido.

Ambas maneras integran componentes motivacionales y estratégicos que las caracterizan. La investigación sobre este constructo en el ámbito universitario permite comprender cómo se produce el aprendizaje, ya que existe una relación entre el enfoque adoptado por el estudiante y los resultados obtenidos tras la ejecución de una tarea (González Geraldo, del Rincon Igea, y del Rincón Igea, 2011).

CAPÍTULO N°2. PROCESOS EDUCATIVOS EN EL NIVEL SUPERIOR DE ENSEÑANZA

En relación a lo expresado en el capítulo anterior, cuando se habla de aprendizaje en el marco de los procesos educativos, se involucra inevitablemente el acto de “enseñar”. Reflexionando acerca de este concepto, no siempre que se intenta enseñar se aprende, y no siempre se aprende cuando el objetivo es enseñar.

Tal como expresa Alicia Fernández (2002) la enseñanza no debe estar centrada únicamente en la transmisión de un determinado cuerpo de conocimientos, sino que es necesaria la recuperación de experiencias y saberes previos de los alumnos, permitir la vivencia de lo que se transmite, la contextualización y concretización de lo mismo. El conocimiento no se puede transmitir directamente en bloque. El enseñante lo transmite a través de una enseña. (...) Se elige una situación, se hace un recorte, se transmite conocimiento y también ignorancia. Además no se transmite, en verdad, conocimiento sino señales de ese conocimiento para que el sujeto pueda, transformándolas, reproducirlo (Fernández, 2002, 2008).

Uno de los principales objetivos de la Educación Superior es formar estudiantes autónomos que sean capaces de dirigir su propia experiencia para generar aprendizaje. Por lo cual existe desde hace un par de décadas, cierta tendencia hacia aprendizaje de tipo constructivista, es decir aquel basado en las propias experiencias del educando (Padilla, 2003).

2.1. Nuevos retos de la Universidad

El contexto económico, político y tecnológico actual le impone permanentemente nuevos desafíos al sistema educativo, el cual debe ser capaz de reestructurar y reorientar sus propuestas pedagógicas a la luz de estos cambios. A la par, la sociedad contemporánea y los estudiantes en formación reclaman nuevas estrategias didácticas

que sean acordes con las exigencias de un mundo que brinda a los individuos una avalancha de información, que no es asimilable de forma sencilla.

Resulta indispensable que las instituciones educativas puedan adaptarse a los requerimientos del mundo actual, formando a las nuevas generaciones en habilidades de pensamiento como complemento de los contenidos específicos que fueron el eje de la enseñanza hasta el momento.

La tendencia actual en el Nivel Superior, propende a que los nuevos profesionales estén capacitados para afrontar una gran variedad de situaciones y problemas que tal vez aún no existen como tal. La Universidad como parte del sistema educativo no puede permanecer ajena a estos desafíos, debiendo capacitar a sus futuros graduados en una formación centrada en prácticas de resolución de problemas, que incentive la creatividad para poder alcanzar un desempeño satisfactorio en el campo profesional (Rosas, Zúñiga, Fernández y Guerrero, 2017).

Uno de los principales objetivos de la Educación Superior es formar estudiantes autónomos que sean capaces de dirigir su propia experiencia para generar aprendizaje. Por lo cual existe desde hace un par de décadas, cierta tendencia hacia aprendizaje de tipo constructivista, es decir aquel basado en las propias experiencias del educando (Padilla, 2003).

Actualmente la educación indica las competencias⁴ como modelo vigente, en donde se establece el desarrollo de habilidades para enfrentar los problemas de forma inteligente y creativa, por tanto, el proceso de enseñanza y aprendizaje debe ser significativo.

⁴ Capacidad de aplicar conocimientos, destrezas y actitudes al desempeño de la ocupación que se trate, incluyendo la capacidad de respuesta a problemas, imprevistos, la autonomía, la flexibilidad, la colaboración con el entorno profesional y con la organización del trabajo.

La brecha entre lo que se espera de los nuevos profesionales y cómo ellos se gradúan de la universidad podría acotarse si se aúnan los esfuerzos para formar a los jóvenes en prácticas de pensamiento desde el comienzo de su formación. En este sentido, podría considerarse que llevar a cabo estas acciones desde el primer año de cualquier carrera universitaria, e incluso desde el Nivel Secundario, también contribuiría a la adaptación y permanencia del alumnado, favoreciendo su desempeño académico durante todo su proceso formativo.

2.2. El tránsito de la Educación Secundaria a la Universidad

En América Latina y en particular en Argentina, existe una marcada heterogeneidad en el estado de preparación y en los niveles de acceso al mundo universitario, resultante de los diversos trayectos educativos realizados por cada estudiante y de la calidad de las ofertas educativas en las escuelas de origen. La articulación entre la Escuela Secundaria y la Universidad necesita mecanismos que faciliten este tránsito y la posterior permanencia del alumno en el Nivel Superior.

En la idea de “la articulación” subyace el concepto de superar compartimientos estancos en educación, de manera que el sistema educativo mejore su calidad, en un marco de diversidad y equidad. La estrategia de articulación requiere una mirada global, integral, reflexiva y sistemática de los actores y de los procesos que intervienen en la organización, estructura y funcionamiento de cada institución.

Los mecanismos de ingreso y permanencia al Nivel Superior deben ser constantemente revisados para ponerlos en contexto y poder así explicitar las fortalezas y debilidades encontradas en un determinado período y grupo poblacional.

En el Nivel Superior se replican muchos de los problemas del Nivel Secundario. El caso de Argentina no escapa a la realidad educativa de América Latina y el Caribe.

Según el informe de la UNESCO-IELSAC (2016) referido al tema, existe un fenómeno generalizado de deserción en la Educación Superior y en su mayoría en los primeros años.

Particularmente en las carreras vinculadas al Área de las Ciencias de la Computación, la problemática recurrente del desgranamiento en los primeros años y la falta de motivación es una constante. La revisión bibliográfica revela al respecto que en muchas universidades del país, se han implementado diversas propuestas para dar respuesta a esta problemática, como por ejemplo: períodos de aprestamiento utilizando herramientas lúdicas, cursos de ingreso con herramientas específicas y prácticas de programación, trabajos interinstitucionales de articulación con el nivel secundario, entre otros. Estas iniciativas lograron resultados prometedores mediante prácticas acotadas a resoluciones concretas y con objetivos orientados en su mayoría al aprendizaje de la programación (Rosas et.al., 2017).

Promover el desarrollo del Pensamiento Computacional desde el Nivel Secundario podría favorecer significativamente las habilidades de los estudiantes para enfrentar y resolver los problemas de índole académicos, personales y sociales. Por Pensamiento Computacional se entiende al conjunto de los procesos de pensamiento implicados en la definición de problemas y en la representación de sus soluciones, de manera que dichas soluciones puedan ser efectivamente ejecutadas por un agente de procesamiento de información (humano, computadora o combinación de ambos). Por lo tanto, en palabras de Rosas et.al., (2017) el pensamiento computacional se basa en resolver problemas haciendo uso de conceptos básicos de la programación. Al momento de resolver un problema, el pensamiento computacional se pone en juego en:

- El *Proceso de Descomposición*. Se descompone un problema complejo (situación o tarea) en pequeños subproblemas, cuyas soluciones combinadas otorgan solución al problema general.

- El *Proceso de Abstracción*. Se centra la atención en las características y rasgos más relevantes del problema, de modo de captar su esencia y filtrar sus características secundarias no fundamentales, para luego crear una representación o modelo simplificado del mismo.

- El *Proceso de Reconocimiento de Patrones*. Se apela al conocimiento de problemas similares resueltos con anterioridad, es decir, se buscan similitudes o patrones de un problema complejo con otro similar ya analizado y resuelto efectivamente. Mientras más patrones se reconozcan, más fácil y rápida será la tarea general de resolver problemas.

- *Algoritmo*. Consiste en un conjunto de instrucciones claras y precisas, que se identifican y se planifican en un determinado orden para la resolución a un problema.

2.3. Los procesos educativos y estrategias de enseñanza

Tal como se viene anticipando, los cambios tecnológicos y el acceso a las Tecnologías de la Información y Comunicación (TIC) están en constante desarrollo y atraviesan nuestras actividades diarias. En este sentido, los estudiantes y profesores están situados en un contexto dinámico y cambiante donde se requiere la puesta en práctica de estrategias de apropiación de los medios.

Estos factores suscitan una revisión permanente de las estrategias de enseñanza, lo cual implica, entre otras cuestiones, contextualizar el marco socio-histórico donde se desarrolla el proceso educativo (Díaz Barriga, 2011) para reflexionar sobre sus prácticas. En las últimas décadas, los aportes de la perspectiva sociocultural de carácter

constructivista se aproximaron al estudio de la cognición situada. De aquí que el aprendizaje se considera una acción humana inseparable del sistema de interacciones sociales situadas en un determinado escenario cultural e histórico (Cabrera y Fariñas, 2005).

Parafraseando a Moreno Castañeda (2011), la docencia se encuentra hoy inmersa en un entorno de complejidad. El autor plantea que el error de algunas instituciones educativas es simplificarla a una profesión u oficio técnico sencillo para el que basta aprender algunas recetas y adquirir destrezas básicas para su desempeño. Esta simpleza suele verse reflejada en los programas oficiales de formación y actualización docente, que suponen como suficiente el aprendizaje de ciertas fórmulas para su generalización a todas las situaciones educativas, que también se toman como ciertas. Sin embargo, la incertidumbre, diversidad y multidimensionalidad ponen de manifiesto la necesidad de adaptar las propuestas a la complejidad de las condiciones educativas que se presentan en la realidad (Castañeda, 2011).

La mirada de Castañeda (2011) parte de la premisa que las modalidades, como variantes del modelo educativo, se distinguen por la diversidad de circunstancias en las cuales suceden los procesos educativos y los recursos que cada una necesita. En este contexto los cambios en el perfil docente no los determinan las modalidades educativas, sino las transformaciones en los modelos educativos y académicos, el perfil de los estudiantes, las condiciones en las que se desarrolla la labor docente y su impacto en las relaciones con el entorno.

En el siglo XXI se presentan docentes que tienen como función principal ayudar a aprender, en un contexto complejo. La complejidad es efectivamente el tejido de eventos, acciones, interacciones, retroacciones, determinaciones, azares, que constituyen nuestro mundo fenoménico (Morin, Ciurana y Motta; 2002). En el plano

educativo, esta complejidad tiene sus propias características relacionadas a las personas y sus interacciones en los procesos educativos, el vínculo con la realidad y el contexto en el que se desarrollan. Esto da lugar a la diversidad, incertidumbre, multidimensionalidad y constantes cambios en el perfil docente.

La diversidad está presente en las personas, los procesos educativos, y en el ambiente en el que se aprende. Hay que atender a la diversidad y a las diferencias personales, y por consecuencia a los modos y medios de aprender.

La incertidumbre se produce en este mismo contexto, donde las diversidades de las personas, el modo de ser, el educar y las experiencias de vida dan lugar a situaciones poco predecibles. Esta situación atraviesa el ser docente y origina cambios, nuevos cuestionamientos que deben ser resueltos y se debe innovar las tareas que se realizan.

La mutidimensionalidad se debe entender a través de la aceptación que las personas no son tablas rasas de una sola dimensión, sino que por el contrario sus vivencias son variadas y los procesos educativos tienen diferentes caras y visiones, aunque desde lo institucional se lo quiera ver parejo, plano y uniforme. La docencia debe realizar múltiples acciones atendiendo a las diferentes dimensiones del aprendizaje: perceptuales, cognitivas y valorativas.

Los docentes están en este cambio y aunque parezcan inmóviles, se terminaran moviendo, y de aquí que los saberes y prácticas deberán transformarse. Es necesario desarrollar una propuesta educativa que incorpore las estrategias necesarias para abordar el cambio.

En el caso puntual de la educación universitaria, en el último tiempo y a nivel internacional, se está trabajando en propuestas innovadoras que incorporen diferentes

usos de la tecnología en el ámbito educativo. Esto ofrece la aparición, revisión y re-significación de diferentes prácticas y modalidades de enseñanza (Casamayor, 2008).

2.4. Rol docente en el proceso de enseñanza

Tal como expresa Ruiz Ortega (2007) en todo proceso renovación en la enseñanza, los docentes son el componente decisorio, pues son ellos los que deben estar convencidos que se necesita de su innovación, de su creación y de su actitud hacia el cambio, para responder no sólo a los planteamientos y propósitos que se fijan en las propuestas didácticas, sino también para satisfacer a las exigencias de los contextos que envuelven a los estudiantes como sujetos sociales, históricos y culturales.

El cambio de paradigma de los modelos didácticos y del rol docente en el proceso de enseñanza y aprendizaje, recoge algunos planteamientos de la teoría asubeliana, al valorar los pre saberes de los estudiantes como aspecto fundamental para lograr mejores aprendizajes. Sin embargo, introduce un nuevo proceso para lograr el cambio conceptual: la enseñanza mediante el conflicto cognitivo. De acuerdo a este modelo el conocimiento científico es incompatible con el conocimiento cotidiano que tiene el educando, lo que plantea como meta, un cambio de sus pre saberes, al mostrarle los alcances y limitaciones de estos, es decir, que se sienta insatisfecho con ellos y que infiera la necesidad de cambiarlos por otros más convincentes (Ruiz Ortega, 2007).

En este sentido se reconoce a un estudiante no sólo con una estructura cognitiva, sino también con unos pre saberes que hace del aprendizaje un proceso de confrontación constante, de inconformidad conceptual entre lo que se sabe y la nueva información. Es entonces, el educando, sujeto activo de su propio proceso de aprehensión y cambio conceptual, objeto y propósito de este modelo.

Siguiendo con la línea argumentativa de Ruiz Ortega (2007), el docente planea las situaciones o conflictos cognitivos, en donde se dé lugar a eventos como la insatisfacción por parte del educando con sus pre saberes, con la presentación de una concepción que reúna tres características para el educando: inteligible, creíble y mucho más potente que sus saberes previos. Desde esta perspectiva, las actividades en el aula deben conducir a los estudiantes a:

- La concientización de la trascendencia de sus pre saberes y la identificación de sus limitaciones.
- La contrastación permanente de lo que sabe con situaciones inteligibles, como requisito para generar el llamado conflicto cognitivo, condición indispensable que desencadena la insatisfacción con los pre saberes y la identificación de teorías más potentes.
- La consolidación de las nuevas teorías o concepciones con mayor poder explicativo, las cuales le permitirán llegar a generalizaciones mucho más inteligibles.

Si bien el cambio conceptual se puede lograr, de manera gradual (Vosniadou, 1992), ya sea por los procesos acumulativos en donde se adicionan nuevas informaciones a los pre saberes del educando o por procesos de cambio en los cuales la transformación de creencias; es importante reconocer en términos de Pozo (1999) que uno de los propósitos en la enseñanza no es sustituir los saberes previos, sino más bien permitir y dar elementos para que el sujeto sea consciente de ellos, los cuestione y distinga dependiendo del contexto en el cual esté desenvolviéndose.

Esta perspectiva plantea una relación directa entre aspectos como los conceptuales, cognitivos, metacognitivos, lingüísticos y motivacionales; en donde se

asuman estos aspectos integrados en los procesos de enseñanza y aprendizaje. Asumir esta perspectiva holística (Tamayo, 2007), hace que se valoren elementos como: experiencias y presaberes del educando, procesos metacognitivos, cognitivos y filosóficos de la ciencia.

2.5. Caracterización de los estudiantes: Generación Z

La mayoría de los universitarios actuales integran el grupo demográfico nacido entre 1994 y 2010, conocido con el nombre de generación Z. Según un informe del diario La Nación (2018) representan el 25,9% de la población mundial y muchos de ellos hoy se están graduando de las universidades para insertarse en el mercado laboral. El rasgo principal que caracteriza a estos jóvenes es que han crecido en la era digital y son conocedores de los nuevos avances tecnológicos, conformando un grupo cien por ciento nativo digital.

Por generación se entiende al conjunto de personas que, por haber nacido en fechas contiguas y haber recibido una educación e influjos culturales y sociales semejantes, adoptan una actitud común en el ámbito del pensamiento y la creación. Sin embargo, esto no significa que sean iguales, ya que algunos se diferencian por el lugar donde viven; por su nivel socio económico y por el acceso a la conexión digital (La Nación, 2018). Es decir, poseen diferentes grados de apropiación y uso de las tecnologías digitales.

En líneas generales, se trata de una generación multitasking que ya no se conforma con ser sujeto pasivo, sino que desea producir el cambio y ser participe desde un rol protagónico. El uso cotidiano de internet modificó sustancialmente su forma de relacionarse, de informarse, de escribir y de hablar. Con el smartphone como el centro de sus vidas, el multitasking se convirtió en un hábito cotidiano: desde ahí planifican,

consultan, buscan, se relacionan y se divierten. Son prácticos, realistas, optimistas, y el empoderamiento es un rasgo que los define (Mociulsky, 2018) (en La Nación, 2018).

Estos jóvenes han vivido experiencias de ampliación de derechos en términos de identidad y género, han crecido en un clima familiar más democrático y diverso, por lo cual tienen incorporado el criterio de libertad y democracia. La digitalización trajo aparejada el fenómeno de la democratización y de la desmonetización, es la primera generación que vivió buena parte de su vida con un teléfono celular inteligente, lo que implica otro tipo de accesibilidad y, por sobre todo, en forma gratuita. Sin embargo, en América Latina las experiencias generacionales siguen teniendo fuertes anclajes en las clases sociales, y muchas de estas tendencias se presentan de forma diferenciada entre los distintos grupos sociales.

No solo son nativos digitales, sino activos globales. Aprendieron a decodificar las redes sociales donde las imágenes son las nuevas vidrieras del siglo XXI, vidriera a la cual ellos mismos se exponen cuándo, dónde y con quiénes quieren. Están atentos a los productos o servicios que sugieren o utilizan sus influencers, quienes muestran sus propias vivencias (Almada, 2018).

La conexión también juega un rol fundamental a la hora de estudiar. Un análisis publicado recientemente por Pearson, compañía multinacional británica de servicios y contenidos educativos, determinó que la mayoría de quienes integran la Generación Z prefiere aprender a través de los videos y tutoriales de YouTube, en lugar de los tradicionales textos impresos.

Los iGen están acostumbrados a tener varios canales de información o estímulo al mismo tiempo. Pueden mirar un video, escuchar el audio de un amigo y a la vez consultar algo en Google, o chequear las nuevas noticias en Instagram. La TV no es un

televisor, sino que lo llaman Smart y es el medio para ver YouTube, Netflix o Amazon Prime.

En cuanto a los medios y dispositivos, el celular es utilizado por el 81% y el TV por el 63% al menos una hora al día en ambos casos. En definitiva, son multipantallas o “agnósticos” en cuanto a las pantallas que usan. Lo que les importa es el contenido, el cual tiene que dar la posibilidad de interactuar, porque esta es una generación activa.

Otra palabra que define a este segmento etario es la flexibilidad. No conciben la rigidez ni las pautas que no tengan para ellos un sentido práctico. Es inconcebible para esta generación, por ejemplo, no tener conectividad durante el horario de trabajo o no poder acceder a las redes sociales, ya que ese es su vínculo con el mundo. Podemos decir que los trabajos de este grupo están insertos en la cuarta Revolución Industrial. Está signado por un cambio en el paradigma del trabajo, que en algunos tiende a desaparecer o transformarse. Hay un nuevo abanico de posibilidades que está relacionado con la gran influencia de la matriz digital, la Big Data y la robotización.

Muchas empresas están adaptando sus procesos y espacios de trabajo de manera que puedan atraer y retener el talento de estos jóvenes. Buscan afianzar en sus equipos ambientes diversos y de colaboración, donde las habilidades tengan un papel preponderante en los procesos de selección.

En definitiva, los estudiantes de hoy son nativos digitales, tienen grandes habilidades para operar con dispositivos tecnológicos, nacieron con un smartphone en sus manos, se les facilitó el uso de una tablet a temprana edad, o tuvieron como herramienta implícita e indispensable un ordenador con conexión a Internet desde sus inicios escolares. Esto lleva a considerar que no se puede desaprovechar lo que traen consigo, por lo cual el trabajo de los profesores consiste en sacar partido de esa energía y capitalizarla, es decir dirigirla hacia el aprendizaje.

CAPÍTULO N°3. LA ROBÓTICA EN LOS PROCESOS EDUCATIVOS.

Uno de los objetivos específicos de la presente tesis radica en el interés de indagar sobre la utilidad de aplicación de robots como herramienta educativa en el Nivel Superior, específicamente en la Universidad Nacional del Noroeste de la Provincia de Buenos Aires.

El proceso de enseñanza y aprendizaje en las instituciones educativas, parte de la necesidad de incursionar en nuevas temáticas relacionadas con la ciencia y la tecnología, donde una de ellas es la robótica, que permite abrir nuevos campos del conocimiento y relacionar otros, mostrando al alumno un rango de posibilidades para su futuro profesional.

Como veremos a lo largo del presente capítulo, la Robótica Educativa se ha desarrollado como una perspectiva de acercamiento a la solución de problemas derivados de distintas áreas del conocimiento como las matemáticas, las ciencias naturales y experimentales, la tecnología y las ciencias de la información y la comunicación, entre otras.

3.1 Robótica

La Robótica representa una de las áreas tecnológicas con más auge en la actualidad, fundamentada en el estudio de los robots, que son sistemas compuestos por mecanismos que le permiten hacer movimientos y realizar tareas específicas, programables y eventualmente inteligentes, valiéndose de conceptos de distintas áreas del conocimiento como la electrónica, la mecánica, la física, la matemática, la electricidad y la informática, entre otras (Pinto Salamanca, Barrera Lombana y Pérez Olguín; 2010).

La integración de estas diferentes áreas del conocimiento, hace posible la obtención de considerables resultados y, dependiendo de su aplicación, los beneficios pueden extenderse más allá del sector industrial y de servicios, alcanzando también las aulas de clase. A este último contexto, la robótica aporta novedosos espacios para el aprendizaje, perfilándose como un nuevo modelo pedagógico que integra la innovación tecnológica y las áreas de conocimiento tradicionales.

3.2 La Robótica Pedagógica

El interés por los aportes de la robótica en los procesos educativos, se evidencia desde los años setenta (Ruiz, 1987), abriendo paso a una nueva disciplina denominada “Robótica Pedagógica”. La Robótica Pedagógica es la actividad de concepción, creación y puesta en funcionamiento, con fines didácticos, de objetos tecnológicos, que son reproducciones reducidas muy fieles y significativas de los procesos y herramientas robóticas usadas cotidianamente en nuestro entorno social, productivo y cultural (Vivet y Nonnon, 1989). En palabras de Ruiz Ortega (2007), esta disciplina permite concebir, diseñar y desarrollar robots educativos para que, desde edades tempranas, los estudiantes se inicien en el estudio de las ciencias y la tecnología, y favorecer así sus procesos cognitivos.

La Robótica Pedagógica utiliza los elementos multidisciplinares de la robótica, con fines didácticos, permitiendo la aplicación de ciertas herramientas tecnológicas, como apoyo en las diferentes metodologías de enseñanza y de aprendizaje, llevando la acción del lugar monopolizado del docente, al universo personal del estudiante (Pinto Salamanca, *et. al*; 2010).

Se plantea entonces a la robótica educativa como una alternativa didáctica que, a la par de los métodos ya establecidos, propende por nuevos enfoques que promuevan en los estudiantes intereses que coadyuven en la creación de ambientes para el aprendizaje

en el que se encuentren circunstancias favorables para la construcción de conceptos y de su interpretación personal de la realidad.

De acuerdo a Monsalves (2011), un ambiente de aprendizaje adecuado para actividades didácticas con robots puede ayudar al aprendizaje de contenidos teóricos. Siguiendo la línea argumentativa de este autor, el uso educacional de la robótica provee a los estudiantes un ambiente de aprendizaje empírico, con un notable potencial para el apoyo del aprendizaje activo. “La robótica forma parte de un enfoque pedagógico centrado en el alumno, que le permite construir objetos tangibles de su propio diseño y con sentido para él.” (Vaillant, 2013, p. 38). Los aprendizajes se construyen a partir de la experiencia del estudiante durante el proceso de construcción y programación de los robots.

La metodología didáctica que contribuye a este tipo de construcción, ha sido la metodología de trabajo grupal en el aula. Las nuevas tecnologías no solo han convertido al aula en un espacio virtual, sino también al trabajo grupal en un trabajo colaborativo.

La mayoría de los autores consultados sobre este tema, coinciden en que para que ocurra un trabajo colaborativo es indispensable que un grupo de personas trabajen juntas y cooperen para alcanzar una meta que de otro modo no podría lograrse. El aprendizaje colaborativo mediado se caracteriza por lo tanto, por contemplar al estudiante en interacción con los demás y no como persona aislada, partiendo del argumento que compartir objetivos y distribuir responsabilidades son formas deseables de aprendizaje (Gutierrez, Yuste y Fustes, 2011). Siguiendo la línea argumentativa de estos autores, este tipo de aprendizaje se caracteriza por potenciar el rol de las herramientas tecnológicas como elementos mediadores en este proceso, facilitando la solución conjunta de los problemas.

Respecto al rol docente en esta metodología de aprendizaje, una vez que la tarea está establecida, se transfiere toda la autoridad a los alumnos siendo estos últimos los responsables de predecir la respuesta (Sanz, Madoz, Zangara y Albanesi, s.f). Se debe tener en cuenta que no basta con poner a un grupo a interactuar para que se produzca el aprendizaje, sino que se deben articular los diferentes elementos para llegar a procesos conjuntos de intercambio y construcción del conocimiento.

Por otra parte y siguiendo con la construcción del conocimiento y adquisición de habilidades por parte del sujeto, y con el espíritu de transformar al aula en un espacio de investigación y cuestionamiento por parte de los estudiantes, donde no se los limita a ejercer un rol pasivo en el que simplemente están “a la espera” de recibir el conocimiento, sino que se los invita a cuestionar, indagar, e ir más allá de lo que el docente puede brindarle; es allí donde toma verdadero sentido incorporar un nuevo concepto de la dinámica educativa, un aprendizaje por indagación, un aprendizaje que se basa en la investigación orientada por el docente. Una dinámica que coloca al alumno en un lugar activo en el que la experimentación tenga un papel protagónico (Couso, Jiménez, López-Ruiz, Mans, Rodríguez, Rodríguez y Sanmartí 2009, 2011).

Bevins y Price (2016) consideran que “haciendo ciencia” se asegura un aprendizaje en términos de aprehensión de conocimientos científicos, afirman que “esta aproximación aporta al alumnado un mayor control del propio aprendizaje y le permite navegar activamente por los caminos que aumentan su comprensión y motivación y mejoran su actitud hacia la práctica científica, incrementando su auto-estima y su capacidad para manejar nuevos datos en un mundo cada vez más complejo” (p. 19).

Sin embargo existen quienes ponen en dudas el éxito de tal premisa. Tal es el caso de lo que algunos investigadores han estudiado en relación a las pruebas PISA. En estos estudios experimentales el predominio de actividades de indagación en las aulas resultó

ser un predictor negativo del rendimiento académico en las pruebas PISA (Areepattamannil 2012; McConney et al. 2014), aunque se encontró una correlación positiva entre el uso de esta metodología y el interés y motivación por las ciencias.

Volviendo a la robótica, a los efectos de la presente investigación, no se la concibe como un fin en sí mismo, sino como una herramienta, es decir como un medio para alcanzar ciertos aprendizajes. Por ello, es preciso distinguir el aprendizaje “de” robótica del aprendizaje “con” robótica. El primero, se concibe en una dinámica de diseño, construcción y explicación de tecnofactos (Gallego y Badillo, 1998), es decir en un ejercicio de creación y construcción de conocimiento. El segundo, en cambio, se asume como el proceso de creación de un contexto (robótica – Informática) que dinamiza y da significado al aprendizaje colaborativo en las diferentes áreas del conocimiento (López Ramírez, 2013). Es decir, el propósito de la robótica educativa se centra en favorecer en los estudiantes el desarrollo de competencias que son esenciales para su desempeño en las carreras del área de informática.

En el marco de la presente tesis, la meta es utilizar la robótica para fomentar el aprendizaje por indagación, orientado a resolver problemas. Tal como expresa Pina (2014), la importancia de emplear un Robot en el aula radica en que los estudiantes se acostumbren a la resolución de problemas, a fomentar su curiosidad, a que los robots les lleven a preguntarse cosas y a construir un aprendizaje significativo.

Entre los beneficios frecuentemente identificados en el uso de la robótica como estrategia didáctica, se destacan: un aumento de la creatividad de los estudiantes, del trabajo en equipo y del autoaprendizaje (Muñoz et.al., s.f). Asimismo, permite el aprendizaje de temas de diferentes áreas de conocimiento, dado el interés que despierta el trabajar con objetos concretos y llamativos como un robot y, si junto con los recursos se implementa una metodología y una adecuada planificación, se estimula en los

estudiantes el aprendizaje de temáticas que, de otra forma, serían más complejas de entender y menos motivantes para su abordaje (López Ramírez, 2013). La robótica como medio o contexto de una educación significativa, facilita a los jóvenes, superar barreras de aprendizaje al despertar su interés y, asimismo, los apoya en el proceso de construir y reconstruir el conocimiento.

Todo lo anterior justifica el uso de robots en el aula para el aprendizaje de conocimientos de diferentes áreas (Pinto Salamanca et.al., 2010). No obstante, una propuesta de robótica educativa para la Educación Superior debe implementarse bajo un enfoque pedagógico que, tal como se ha desarrollado en los capítulos anteriores, tenga en cuenta el ambiente de aprendizaje, los estilos de aprendizaje, la planeación de las actividades, los recursos y la metodología a desarrollar. Estas características del modelo pedagógico garantizarán la construcción y reconstrucción del conocimiento por parte del estudiante (López Ramírez, 2013).

3.3 El enfoque pedagógico

Las actividades generadas para la construcción del conocimiento utilizando la robótica, se caracterizan por la aplicación de teorías pedagógicas como el constructivismo de Piaget, que plantea que el conocimiento no es simplemente transmitido por el profesor al estudiante, sino que es adquirido activamente por quien aprende; y el construccionismo de Papert, (en Xudong y Weinbeg, 2003), que sugiere que quienes aprenden están particularmente motivados cuando viven la experiencia de construir, ya sea un robot, un tecnofacto, un poema, un castillo de arena, un programa de computador o una teoría científica, sobre lo cual puedan reflexionar y compartir con otros.

En definitiva, como expresa Papert (1995), el construccionismo tiene como principal característica que observa la idea de la construcción mental más cerca que los

otros -ismos educativos. La aplicación de estas teorías de aprendizaje tiene como propósito desarrollar el pensamiento sistémico (Andrade, 2007), el desarrollo cognitivo, el desarrollo del pensamiento científico y la capacidad creativa e investigativa en los estudiantes.

Para promover el enfoque pedagógico, la informática aporta el uso de herramientas tecnológicas, por ejemplo, la manipulación de robots móviles en ambientes virtuales, como en Pérez (2004), donde se realiza la experiencia con estudiantes desde los 5 hasta 12 años de edad, con el fin de promover en ellos los diferentes tipos de pensamiento y su desarrollo cognitivo, aplicando teorías de aprendizaje como el constructivismo, la holística y la metacognición.

Asimismo, se destaca la importancia de la robótica en la integración de las diversas áreas del conocimiento y la aplicación del pensamiento sistémico para la creación de entornos de aprendizaje que proporcionen un alto grado de motivación. Es posible entonces utilizar la robótica como elemento motivador e integrador del conocimiento (Bers y Urrea, 2000). Involucrando la robótica como motor de innovación, se logra un cambio en la forma de pensar y actuar tanto de los profesores como de los estudiantes.

El cumplimiento de estas cuestiones requiere de transformaciones profundas en el sistema escolar, iniciando en los Ministerios de Educación, que deben dar solución a tres problemas fundamentales: a) la falta de lineamientos pedagógicos y metodológicos que orienten a los docentes sobre el aprendizaje con robótica, ya que no existen estándares para el área de tecnología e informática, ni se encuentran propuestas estructuradas para trabajarla en la escuela, b) la necesidad de formación docente para utilizar los recursos y aplicar una determinada metodología para el aprendizaje de los conceptos básicos de robótica, y c) la falta de infraestructura y recursos; ya que se

necesita dotar a las instituciones educativas de los recursos necesarios para poder implementar sistemas y llegar hasta la automatización de los mismos, ya que se encuentran con el gran inconveniente de los altos costos de los elementos necesarios para implementar la automatización de procesos.

Para concluir, los robots, entendidos estos como una máquina automática o autónoma que posee cierto grado de inteligencia, capaz de percibir su entorno y de imitar determinados comportamientos del ser humano (Juárez Landín, 2016), cuentan con el potencial de facilitar el aprendizaje de un lenguaje de programación, propiciar la experimentación y estimular las competencias asociadas a la resolución de problemas (Willging, Astudillo, Castro, Bast, Occelli y Distel, s.f). Para que los robots funcionen con autonomía, es necesario indicarles qué tienen que hacer ante cada situación, y para eso, hace falta compartir un idioma como el lenguaje de programación que permite la interacción mediante las órdenes que se les pueden dar para realizar acciones específicas (Juárez Landín, 2016).

CAPÍTULO 4. PROGRAMACIÓN

Tal como se expuso en el capítulo anterior, los robots cuentan con la virtud de facilitar el aprendizaje de un lenguaje de programación, propiciar la experimentación y estimular las competencias asociadas a la resolución de problemas (Willging, Astudillo, Castro, Bast, Occelli y Distel, s.f). Para que los robots funcionen con autonomía, es necesario indicarles qué tienen que hacer ante cada situación, y para ello, hace falta compartir un idioma como el lenguaje de programación que permite la interacción mediante las órdenes que se les pueden dar para realizar acciones concretas.

Cuando una persona programa lo que hace básicamente es diseñar una solución para un problema determinado a través de algoritmos, y como no existe forma de determinar una única solución tampoco existe manera de establecer que un algoritmo sea el único que soluciona tal problema. De este modo, esa “libertad” con que se pueden pensar diferentes maneras de resolver problemas indefectiblemente pone en juego la creatividad de la persona, en este sentido está dada la utilidad de la programación.

Es así como llegamos al último capítulo del primer apartado de esta tesis donde se pretende introducir al lector en la importancia del aprendizaje de programación de computadoras, no solo en términos académicos (por las capacidades, destrezas y competencias que se desarrollan) sino fundamentalmente en materia profesional y laboral.

4.1. Uso versus comprensión

Para comenzar este capítulo se analiza la diferencia entre el uso de una computadora y la comprensión de su funcionamiento. En los tiempos que corren es habitual que muchas personas sepan utilizar un ordenador o cualquier otro artefacto tecnológico, sin antes haber sido formados en estas habilidades; de hecho, los

conocimientos sobre cómo operar una computadora escapan a la educación formal. Pero ¿qué sucede cuando se habla de comprensión? es decir, se utiliza Whatsapp, pero ¿sus usuarios conocen cómo opera internamente esta aplicación? o se publica en Instagram, pero ¿se sabe cómo se programó esta aplicación? Dicho en otros términos, existe una brecha entre el uso de un software y la comprensión de su operación o, mejor aún, la comprensión de la lógica que maneja ese software.

Cuando se habla de saber utilizar un ordenador se está haciendo referencia a los conocimientos básicos de manejo de una computadora y a las habilidades referidas al uso de los programas que habitualmente se utilizan en una oficina, por ejemplo: trabajar con archivos y carpetas, instalar y desinstalar programas, emplear un editor de texto o una planilla de cálculo, navegar por Internet, enviar emails, y quizá hasta operar una base de datos sencilla o diseñar una presentación como material de apoyo de una exposición oral. Estas aptitudes, y otras tantas, son necesarias para algunos ámbitos laborales, como aquellos que específicamente requieran de los conocimientos y destrezas de un operador de PC. Sin embargo no es la creatividad la potencialidad que aquí se despliega.

La demanda laboral actual, sobre todo para el área de las ciencias de la computación, se orienta a la búsqueda de recursos humanos que puedan realizar tareas en las que se involucre la creatividad, es decir la capacidad para “dar” nacimiento a una idea o innovar una ya existente. Es innegable que el mundo tecnológico se encuentra en constante desarrollo y ello promete que quienes adquieran habilidades y conocimientos en esta área pueden llegar a tener más posibilidades de crecimiento profesional que en otras carreras.

En definitiva, si bien resulta útil adquirir conocimientos y destrezas en cuanto a la utilización de un ordenador, esta “utilidad” está más relacionada con una demanda

laboral pasada que con los propios intereses de crecimiento individual de quienes componen hoy una sociedad. La humanidad necesita individuos creativos, innovadores, que generen y produzcan, y no personas que se limiten a responder a ciertas demandas.

Cuando una persona programa lo que hace básicamente es diseñar una solución para un problema determinado a través de algoritmos, y como no existe forma de determinar una única solución tampoco existe manera de establecer que un algoritmo sea el único que soluciona tal problema. De este modo, esa “libertad” con que se pueden pensar diferentes maneras de resolver problemas indefectiblemente pone en juego la creatividad de la persona, en este sentido está dada la utilidad de la programación.

4.2. Programación y mercado laboral

Tomando en consideración el contexto mundial respecto del avance tecnológico, se puede inferir que aprender a programar, como una de las tantas habilidades tecnológicas que pueden desarrollarse, abrirá muchas puertas en el mundo laboral. Veamos a modo de ejemplo algunos portales de Argentina en lo que concierne a la oferta laboral actual.

En el portal “Buscojobs Argentina”⁵, que se especializa en la intermediación de la oferta y demanda de empleo, las empresas especifican sus requerimientos, mientras que las personas completan sus currículums y de este modo se conocen y vinculan. El portal se visualiza del siguiente modo:

Para la categoría “Tecnologías de la Información” Se encontraron 5.648 ofertas laborales al 10 de enero del 2019, ubicándose en el segundo puesto del ranking de ofertas laborales (observar la Ilustración 1).

⁵ buscojobs.com.ar

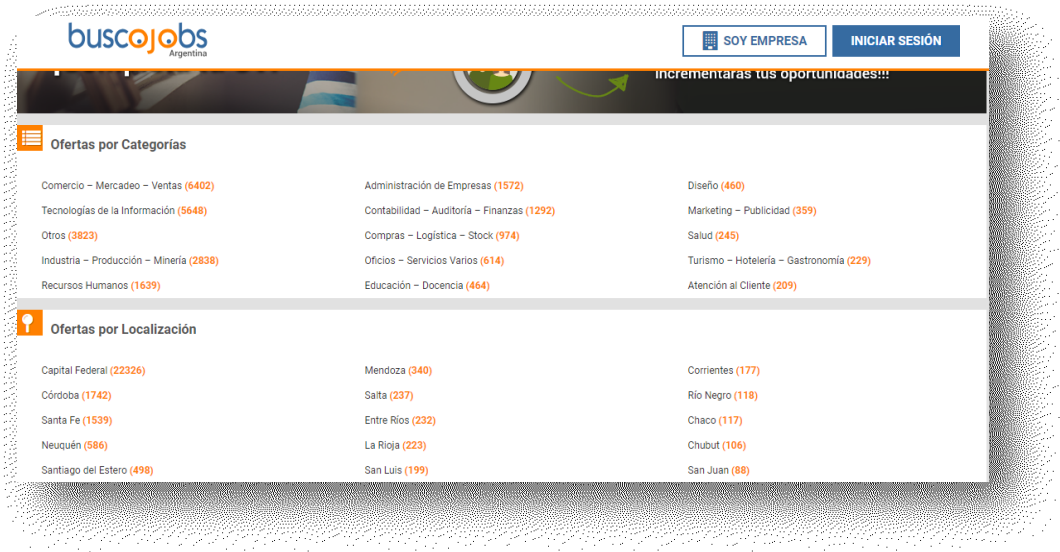


Ilustración 1: Ofertas por categoría y por localización.

Luego se puede observar en la Ilustración 2, la primera subcategoría “Programación”, con una cantidad de ofertas laborales de 2.493. Los datos demuestran que la cantidad de puestos de trabajos que se requieren en el área, representa un número significativo en nuestro país.

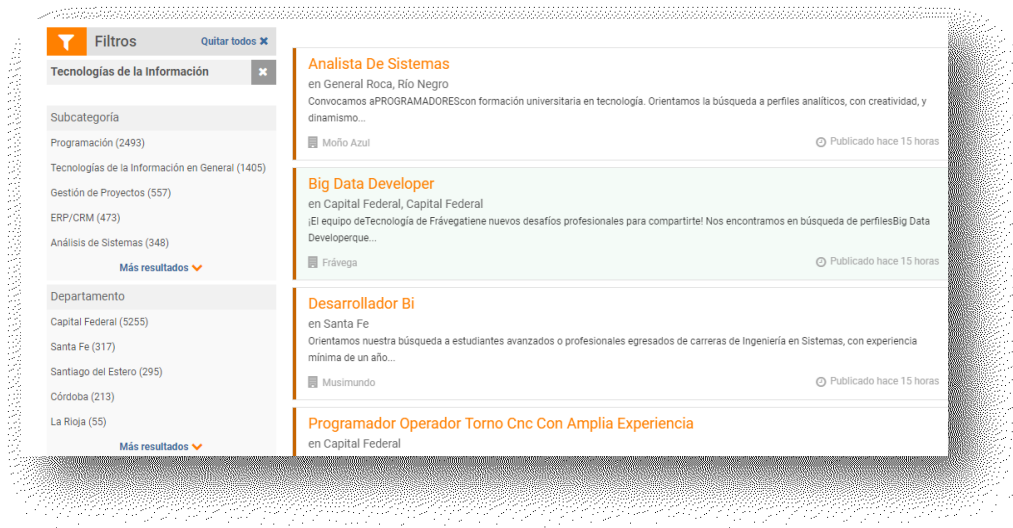


Ilustración 2: Subcategoría “Programación”.

Por otra parte en el portal “CompuTrabajo”⁶, bolsa de trabajo con ofertas en Argentina que permite realizar una amplia búsqueda por filtros de ubicación (provincia, ciudad y barrio), categoría, tipo de contrato, tipo de jornada y salario; se encuentran ciertos datos como pueden observarse en la Ilustración 3 a continuación:



Ilustración 3: Ofertas laborales en CompuTrabajo filtrada por la palabra clave “Informática/Telecomunicaciones”

Como se puede apreciar al 10 de enero de 2019 hay 2.853 ofertas laborales relacionadas con la informática y las telecomunicaciones.



Ilustración 4: Ofertas laborales en CompuTrabajo filtrada por la palabra clave “Programador”

⁶ www.computrabajo.com.ar

Para la misma fecha hay 316 ofertas laborales relacionadas con la palabra clave “Programador” (observar la Ilustración 4).

Al consultar otro portal, en este caso “Zonajobs”⁷, se puede observar que se centran en Argentina y se organizan por ubicación, áreas de trabajo, posición de trabajo y fechas de publicación.

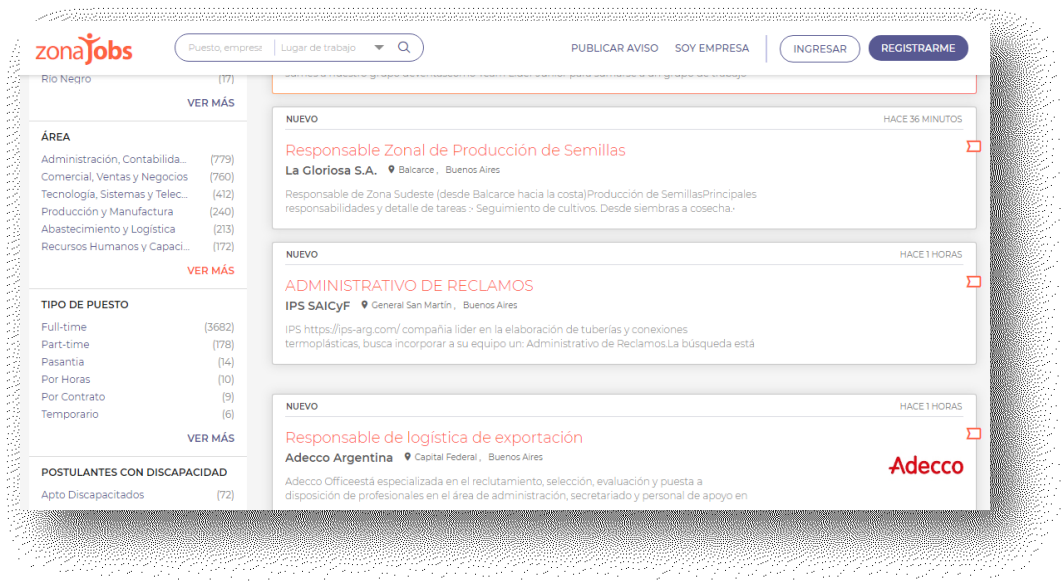


Ilustración 5: Captura de pantalla del portal ZonaJobs

⁷ www.zonajobs.com.ar

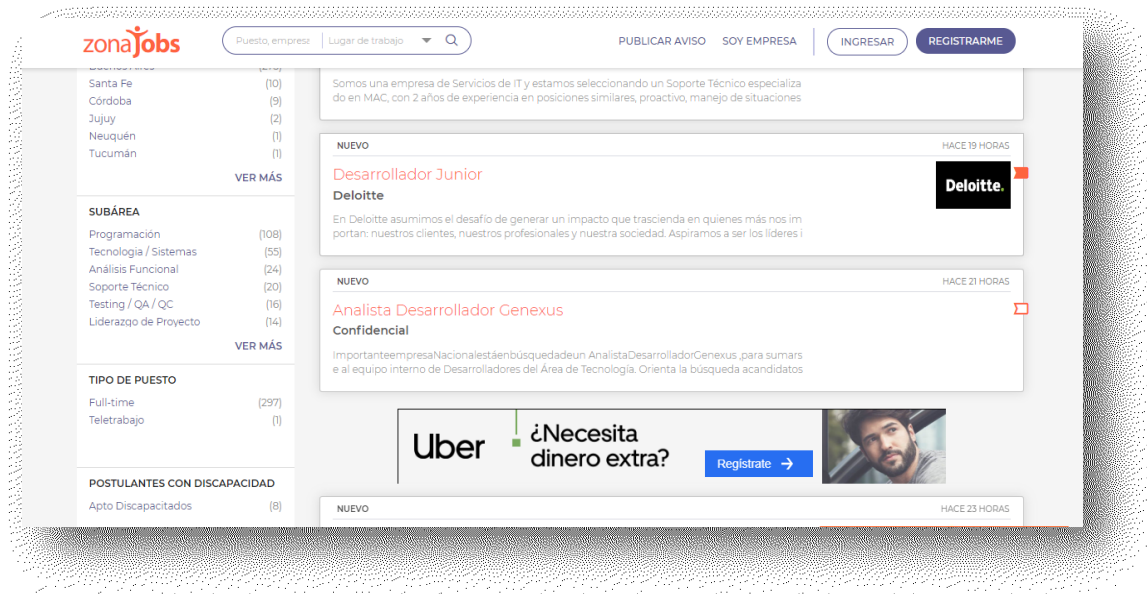


Ilustración 6: Filtrado por área "Tecnología"

Para la subárea “Programación”, como puede observarse en la Ilustración 6, hay una oferta de 108 puestos laborales; mientras que para la subárea “Tecnología/Sistemas” hay una oferta de 55 puestos posicionándose en la primera y segunda subáreas en el portal, sobre un total de 299 trabajos de Tecnología.

Siguiendo con “Bumeran”⁸, como portal especializado en búsquedas de empleo en Argentina, si se filtra por el área “Tecnología” nos encontraremos con sub áreas del siguiente modo: en “Programación” hay 224 puestos ofrecidos, para “Tecnología/Sistemas” hay 143, “Sistemas” 141, para “Soporte Técnico” hay 62, y así en decremento con otras sub áreas (observar la Ilustración 7).

⁸ www.bumeran.com.ar

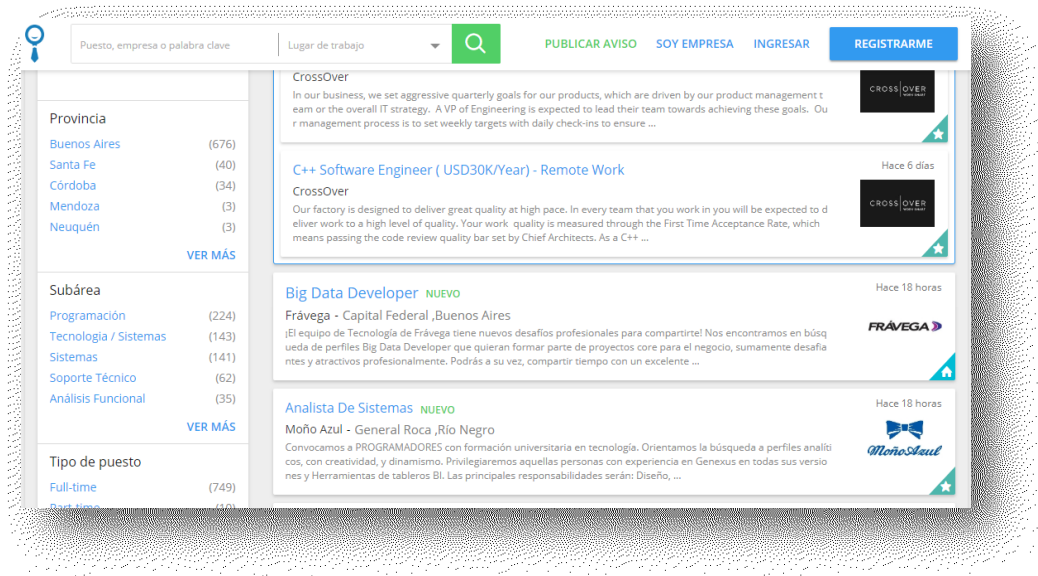


Ilustración 7: Portal Bumeran filtrado por el área "Tecnología"

Los portales nombrados son algunos de los tantos sitios en los que se pueden hallar ofertas laborales para todas las categorías, observándose que para las áreas y sub áreas que se están analizando en este trabajo se requieren habilidades tecnológicas varias.

En ese sentido es válido inferir que la demanda laboral que se genera a raíz de este crecimiento tecnológico va en aumento. Además cuando se aprehenden habilidades y destrezas relacionadas con la programación, la ganancia individual es aún mayor dado que en estos casos las competencias que se desarrollan son mucho más significativas.

En palabras de Zapata-Ros:

Las competencias de codificar son la parte más visible de una forma de pensar que es válida no sólo en ese ámbito de la actividad mental, la que sostiene el desarrollo y la creación de programas y de sistemas. Hay una forma específica de pensar, de organizar ideas y representaciones, que es terreno abonado y que favorece

las competencias computacionales. Se trata de una forma de pensar propicia para el análisis y la relación de ideas, para la organización y la representación lógica. Esas habilidades se ven favorecidas con ciertas actividades y con ciertos entornos de aprendizaje desde las primeras etapas. Se trata del desarrollo de un pensamiento específico: el pensamiento computacional. (Zapata Ros, 2015, p. 1).

En relación a lo anterior se ha comprobado que el pensamiento computacional ayuda a desarrollar importantes habilidades que van más allá de la formación en la disciplina misma, destacándose:

- La descomposición en subproblemas. El hecho de poder dividir tareas complejas en tareas de menor complejidad, le permite al sujeto concentrarse en problemas más concretos y sencillos de resolver.
- La generalización de casos particulares.
- La abstracción de problemas.
- La modelización de situaciones.
- Los procesos de diseño, implementación y prueba.

En este sentido, incorporar contenidos que involucren el pensamiento computacional puede llegar a ser muy positivo.

4.3. La importancia de la enseñanza de la programación en las Escuelas

“There's a lot to learn when it comes to coding. How do you know where to start? Don't worry, we've got you covered. We have something for everyone, whether you're new to coding, or a long time learner, explore the options below

for #HourOfCode with #Codecademy.”Si se traduce el texto sería algo como lo siguiente: “Hay mucho que aprender cuando se trata de codificación. ¿Cómo sabes por dónde empezar? No te preocupes, te tenemos cubierto. Tenemos algo para todos, ya sea para principiantes como para quienes tengan experiencia, explora las siguientes opciones para #HourOfCode con #Codecademy”

Éste párrafo se puede visualizar en la web de la CodeAcademy⁹ que, junto con Code.org¹⁰, buscan suscitar la enseñanza de la programación en escuelas. Si bien la iniciativa pone énfasis en las grandes posibilidades laborales que se abren para quienes sepan programar, también se vislumbran otros beneficios a partir de trabajar el pensamiento computacional. Se ha estudiado e investigado a lo largo de todo el mundo acerca de las capacidades, destrezas y competencias desarrolladas por estudiantes al aprender a programar.

En los años 80 se agregó en la currícula escolar, la enseñanza de programación con el lenguaje de programación Logo¹¹. “El Logo Programming Language, un dialecto de Lisp¹², fue diseñado como una herramienta para el aprendizaje.

Las actividades de programación del logotipo son matemáticas, lenguaje, música, robótica, telecomunicaciones y ciencia. Se usa para desarrollar simulaciones y crear presentaciones multimedia y juegos. El logotipo está diseñado para tener un "umbral bajo y sin límite máximo": es accesible para principiantes, incluidos los niños pequeños, y también admite exploraciones complejas y proyectos sofisticados por usuarios experimentados.

⁹ www.codecademy.com

¹⁰ Code.org es una organización sin fines de lucro dedicada a expandir la participación de la educación en Ciencias de la Computación llevándola a más escuelas. A su vez, busca aumentar la participación en esta disciplina de las mujeres y otros grupos sociales que se encuentran actualmente subrepresentados.

¹¹ www.el.media.mit.edu/logo-foundation

¹² Lisp es un lenguaje de programación que muestra la información en forma de listas, de ahí el nombre (List-Processing). Fue creado básicamente para el procesamiento de listas.

Los entornos de Logo más populares han involucrado a la Tortuga, originalmente una criatura robótica que se sentaba en el piso y podía ser dirigida para moverse escribiendo comandos en la computadora.

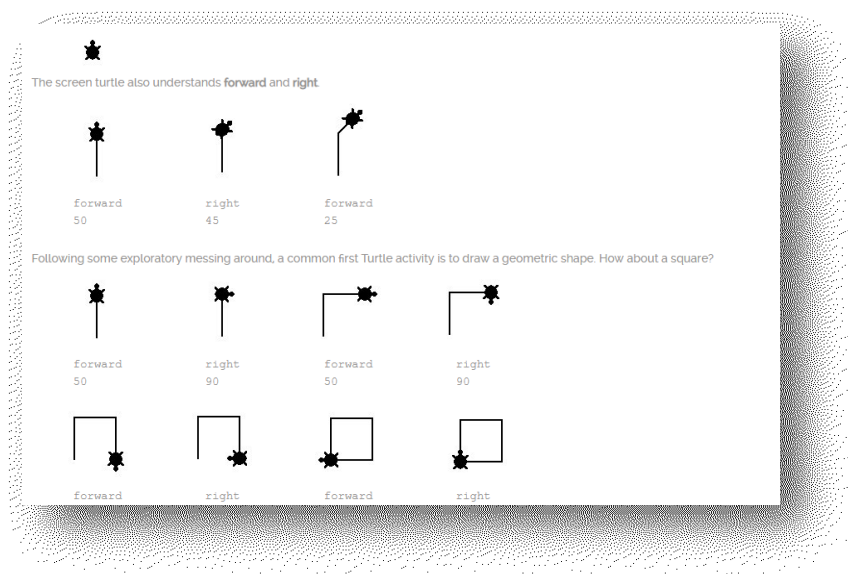


Ilustración 8: captura de pantalla del sitio "Logo Foundation"

En la *Ilustración 8* puede apreciarse cómo indicarle a la tortuga qué puede hacer: ir hacia adelante (`forward x`), girar hacia la derecha (`right x`), etc. Pronto, la Tortuga migró a la pantalla de gráficos de computadora donde se usa para dibujar formas, diseños e imágenes.

Esta iniciativa de insertar el lenguaje Logo en las escuelas captó la atención de diferentes grupos de investigación que estudiaron otras habilidades que se aprendían a la hora de trabajar con un lenguaje de programación.

En el artículo "*Effects of Logo and CAI environments on cognition and creativity*", Clements (1986) comenta acerca de evaluaciones sobre los efectos de aprender programación de computadoras e instrucción asistida por computadora en habilidades cognitivas específicas, habilidades metacognitivas, creatividad y logros. El artículo refiere a la evaluación de 72 niños de 6 a 8 años, en los que se midió el nivel de

la competencia operativa, la creatividad y el rendimiento antes de aprender a programar. Luego, en los exámenes posteriores, se pudo revelar que las calificaciones fueron significativamente más elevadas, lo que permitió concluir que quienes usaron Logo en educación infantil demostraron mayor capacidad de atención, más autonomía, y mostraban un mayor placer por el descubrimiento de nuevos conceptos (Clements, 1986).

PARTE III. ENCUADRE METODOLÓGICO.

TIPO DE ESTUDIO

Conforme a los objetivos internos, es decir al tipo de información que se desea obtener, la presente investigación se define como explicativa. Sin embargo, es importante señalar que estas investigaciones muchas veces involucran en sí mismas los propósitos de otros estudios; como ser exploratorios, descriptivos y correlacionales.

Dicho en otros términos, esta investigación no se sitúa únicamente como explicativa, sino que contiene elementos propios de los estudios exploratorios y de los estudios descriptivos. Veámoslo de un modo ilustrativo: el estudio se desarrolla inicialmente como explicativo ya que, tal como su nombre lo indica, aspira a explicar por qué dos o más variables están relacionadas entre sí (Hernández Sampieri, 2014), es decir se orienta a establecer la relación entre la robótica educativa y el aprendizaje de la Programación Imperativa, y por consiguiente su incidencia en la disminución del índice de deserción y desgranamiento de los estudiantes de primer año de las carreras del área de informática de la UNNOBA.

Sin embargo, al revisar la literatura disponible la investigadora no encuentra antecedentes que puedan aplicarse a su contexto, por lo cual comienza a explorar el fenómeno, es decir, comienza a incursionar en el uso de robots como herramienta educativa en el nivel superior, por lo cual la investigación toma ahora un camino exploratorio, al examinar un tema o fenómeno que no ha sido abordado con anterioridad y que es relativamente desconocido (Hernández Sampieri, 2014). De esta manera se plantea un estudio que brinde un contexto de aplicación adecuado al uso de una herramienta pedagógica poco estudiada como el Robot Educativo Programable.

Finalmente se describe el fenómeno con mayor exactitud dando como resultado final nuevas estrategias educativas dentro del paradigma imperativo de la programación

de computadoras en la UNNOBA, utilizando como herramienta fundamental el Robot Educativo Programable (REP).

Siguiendo con la clasificación, según su alcance temporal o diacrónico, la investigación se clasifica como longitudinal retro prospectiva y según sus fuentes como mixta, al emplearse tanto fuentes de primera mano (cuestionarios, escalas, evaluaciones, etc.) como así también informaciones que ya han sido producidas por otras personas con otros fines (archivos, libros, pdf.). El tratamiento de estas últimas se realiza mediante el análisis crítico de toda la información adquirida, para ampliar conocimientos y argumentar teóricamente el trabajo.

DISEÑO DE LA INVESTIGACIÓN.

Una vez definido el tipo de estudio a realizar y establecidas las hipótesis de la investigación, la autora de la presente tesis concibe la manera práctica y concreta de responder a los objetivos de este estudio. Con tal finalidad se selecciona un diseño de investigación para aplicarlo al contexto particular del estudio.

Esta investigación se clasifica como experimental “pura”, ya que en esta se manipulan deliberadamente una o más variables independientes (supuestas causas) para analizar las consecuencias que esta manipulación tiene sobre la variable dependiente (supuesto efecto), dentro de una situación de control. La variable independiente (representada por el REP, como así también por las estrategias didácticas que lo enmarcan) resulta de especial interés para la investigadora, por ser la variable que se hipotetiza, que será una de las causas que producen el efecto supuesto. Para obtener evidencia de esta supuesta relación de causalidad, se manipulará la variable independiente y se observa si la dependiente varía o no (Hernández Sampieri, 2014).

La variable independiente será manipulada en este caso conforme al criterio de presencia-ausencia. Es decir se expone al grupo experimental a la presencia de la variable independiente (REP), para observar sus resultados y compararlos con los obtenidos en los grupos de control, es decir en aquellos que no fueron expuestos al estímulo experimental (estudiantes que cursaron la asignatura durante el periodo 2005-2017)

El efecto que el tratamiento experimental (variable independiente) tiene sobre la variable dependiente se medirá a través de cuestionarios, escalas, evaluaciones y observación.

En cuanto al control o validez interna de la situación experimental, se estima que la variación de las variables dependientes (aprendizaje, motivación entre otras) se debe a la manipulación de la variable independiente (REP) y no a otros factores o causas. Entonces, para lograr el control del experimento se controlará la influencia de otras variables extrañas en las variables dependientes, para poder saber a ciencia cierta si existe relación de causalidad (Hernández Sampieri, 2014). Por ello para lograr esto se plantea que los grupos difieran entre sí solamente en la exposición al REP y las estrategias didácticas que acompañan su implementación.

UNIVERSO.

Es la totalidad del fenómeno a estudiar, el conjunto de individuos o elementos de los que se desea conocer algo en una investigación y que poseen una propiedad común, susceptible de ser estudiada. En el presente estudio el universo está constituido por todos los estudiantes de primer año de las carreras del área de informática de la UNNOBA de la ciudad de Junín. Se realizará un estudio de caso, con los estudiantes

que en los años 2018 y 2019 se encuentren cursando la asignatura Introducción a la Programación Imperativa, los cuales hacen un total de 138 y de 216 respectivamente.

A continuación pueden observarse en las ilustraciones 9, 10 y 11; los inscriptos en los años 2018 y 2019, para ambas sedes.

SIUGUARANI
MÓDULO DE GESTIÓN ACADÉMICA

PAULA LENCINA

Cursadas Clases Exámenes **Reportes** Derechos Perfil: Docente

Inscriptos a comisiones

Filtro

Período lectivo: 2018 - 1° cuatrimestre (G) Actividad: Todas
 Docente: LENCINA, PAULA (390) Ubicación: JUNIN

2018 - 1° cuatrimestre (G)
Introducción a la Programación Imperativa (00863)

Comisión	Subcomisión	Ubicación	Responsabilidad	Turno	Inscripciones
COM 1 JUNÍN	No tiene	JUNIN	Adjunto	Sin definir	42
COM 2 JUNÍN	No tiene	JUNIN	Adjunto	Sin definir	48
COM 3 JUNÍN	No tiene	JUNIN	Adjunto	Sin definir	48

Ilustración 9: inscriptos a la cursada de IPI en la sede de Junín, año 2018

SIUGUARANI
MÓDULO DE GESTIÓN ACADÉMICA

María Teresa LANZILLOTTA

Cursadas Clases Exámenes **Reportes** Derechos Perfil: Docente

Inscriptos a comisiones

Filtro

Período lectivo: 2018 - 1° cuatrimestre (G) Actividad: Todas
 Docente: LANZILLOTTA, María Teresa (330) Ubicación: Todos

2018 - 1° cuatrimestre (G)
Introducción a la Programación Imperativa (00863)

Comisión	Subcomisión	Ubicación	Responsabilidad	Turno	Inscripciones
COM 1 PERG	No tiene	PERGAMINO	Adjunto	Sin definir	62
COM 2 PERG	No tiene	PERGAMINO	Adjunto	Sin definir	17

Ilustración 10: inscriptos a la cursada de IPI en la sede de pergamino, año 2018

SIUGUARANI MÓDULO DE GESTIÓN ACADÉMICA

PAULA LENCINA

Reportes

Inscritos a comisiones

Filtro

Período lectivo: 2019 - 1° cuatrimestre (G) Actividad: Todas

Docente: LENCINA, PAULA (390) Ubicación: PERGAMINO

2019 - 1° cuatrimestre (G)

Introducción a la Programación Imperativa (00863)

Comisión	Subcomisión	Ubicación	Responsabilidad	Turno	Inscripciones
PERG COM 1	No tiene	PERGAMINO	Adjunto	Sin definir	37
PERG COM 2	No tiene	PERGAMINO	Adjunto	Sin definir	40

Ilustración 11: inscriptos a la cursada de IPI en la sede de Pergamino, año 2019

SIUGUARANI MÓDULO DE GESTIÓN ACADÉMICA

María Teresa LANZILLOTTA

Reportes

Inscritos a comisiones

Filtro

Período lectivo: 2019 - 1° cuatrimestre (G) Actividad: Todas

Docente: LANZILLOTTA, María Teresa (330) Ubicación: Todos

2019 - 1° cuatrimestre (G)

Introducción a la Programación Imperativa (00863)

Comisión	Subcomisión	Ubicación	Responsabilidad	Turno	Inscripciones
COM 1 JUNÍN	No tiene	JUNÍN	Adjunto	Sin definir	66
COM 2 JUNÍN	No tiene	JUNÍN	Adjunto	Sin definir	73

Ilustración 12: inscriptos a la cursada de IPI en la sede de Junín, año 2018

Debido a que el universo no posee grandes dimensiones se procede a estudiar su totalidad, motivo por el cual no se seleccionará una muestra.

INSTRUMENTOS DE RECOLECCIÓN DE DATOS.

Se emplean las siguientes técnicas e instrumentos de recolección de datos: cuestionarios autoadministrados, escalas, observación y evaluaciones. La observación constituye un método de recopilación de información primaria mediante la percepción

directa de los elementos del fenómeno estudiado que se consideran significativos para los objetivos de la investigación. Las escalas consisten en un conjunto de ítems presentados en forma de afirmaciones o juicios, ante los cuales se solicita la reacción de los participantes (Hernández Sampieri et al., 2014). La encuesta por cuestionario consiste en la estructuración de un conjunto de preguntas, a través de un instrumento estandarizado, para obtener la información necesaria sobre las variables en estudio (Sabino; 2000). Y la evaluación consiste en un instrumento de reflexión y análisis sobre las acciones realizadas y el logro de los objetivos propuestos, que se enmarcan en valorizar si los estudiantes logran apropiarse de los conceptos básicos de la programación imperativa, resolver problemas mediante un algoritmo, modularizar y parametrizar y aplicar el paradigma estructurado en la resolución de problemas y correctas estructuras de control.

PARTE IV. EL CASO DE LA UNNOBA.

Este apartado se subdivide en seis capítulos o secciones, a saber: 1. *antecedentes*, donde se presenta la problemática de la deserción y el desgranamiento estudiantil en las carreras del área de Informática de la Escuela de Tecnología (ET) de la UNNOBA. 2. *Descripción de la asignatura*, para caracterizar el caso particular de la cátedra “Introducción a la Programación Imperativa” y mostrar la necesidad de desarrollar metodologías para contribuir a mejorar el aprendizaje de los estudiantes en la asignatura; 3. *Metodología propuesta*, donde se presentan las estrategias, técnicas y herramientas a incorporar en el marco de la asignatura en cuestión; 4. *Marco Referencial*, donde se describe el ámbito o contexto institucional en el cual se llevará a cabo la experimentación (recursos materiales, técnicos, físicos, humanos, profesionales). 5. *Experimento*, para explicar los diferentes momentos en la aplicación de la metodología durante el período 2018- 2019; y finalmente 6. *Discusión de los resultados*, donde se realiza un análisis pormenorizado de los datos arrojados por las encuestas, la experimentación propiamente dicha y la evaluación. Estos datos son interpretados a la luz de la teoría que sustenta la presente tesis.

CAPÍTULO N° 1. ANTECEDENTES

1.1. La problemática de la deserción y el desgranamiento

La Escuela de Tecnología es una de las Unidades Académicas de la UNNOBA y comprende las carreras de las áreas de Ingeniería, Diseño e Informática. Dentro del área de Informática se dictan cuatro carreras: Ingeniería en Informática, Licenciatura en Sistemas, Analista de Sistemas y Tecnicatura Universitaria en Soporte Informático. El desgranamiento y la deserción estudiantil se presentan como problemáticas comunes a las carreras de esta área, sobre todo en los dos primeros años.

Con el objetivo de determinar esta problemática e identificar sus factores causales, la ET lleva a cabo un relevamiento de datos de los estudiantes inscriptos, matriculados y graduados, cuatrimestre por cuatrimestre, de acuerdo a cada cohorte. Es decir, se realiza un análisis longitudinal y transversal de la totalidad de los años de cursada para establecer el índice de deserción por cada carrera y determinar su momento más crítico.

Para concretar dicho estudio se consideran los siguientes indicadores:

- *Índices de deserción por cohorte*: es el resultado de las diferencias, para cada cohorte, entre el número total de estudiantes que ingresan a la carrera y el número que culmina la cursada del plan de estudio. Al respecto es preciso señalar que si bien el número de ingresantes ha aumentado considerablemente en los últimos ocho años, esta cifra no se corresponde con el número total de graduados.

- *Índices de deserción por año/nivel de cada carrera*: es el número total de desertores de cada año/nivel de una carrera, sobre el total de matriculados en el mismo año/nivel. En las carreras del área de Informática, los mayores índices de deserción se presentan durante los dos primeros años de estudio (observar las ilustraciones 13 y 14).

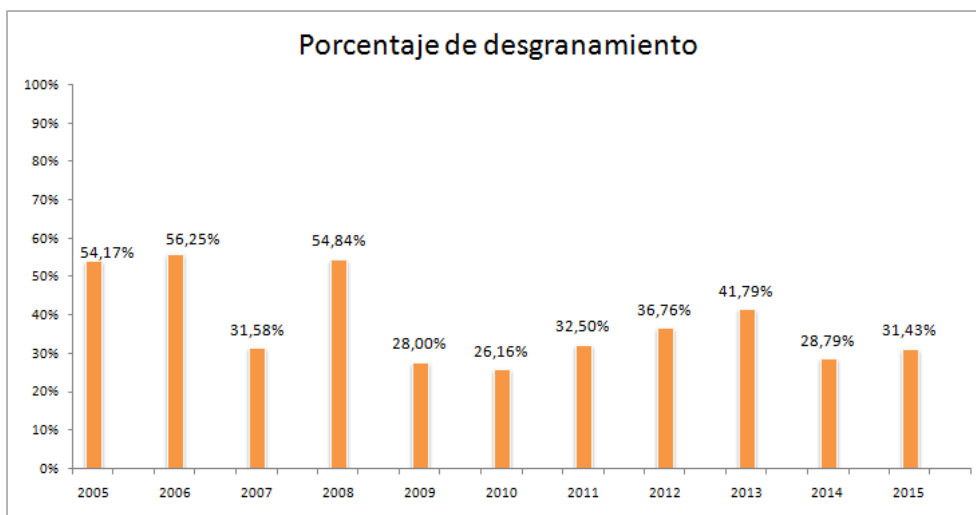


Ilustración 13: Índices de deserción para la carrera “Licenciatura en Sistemas”

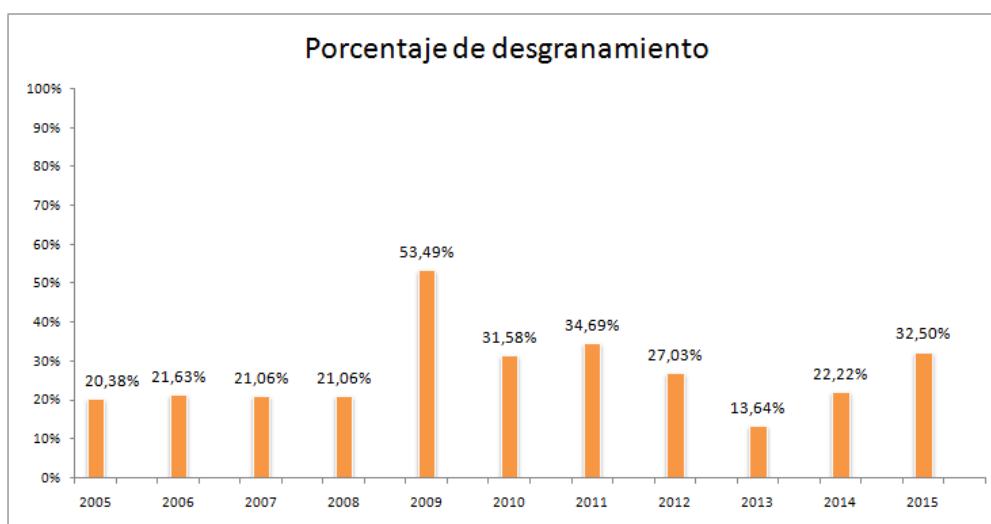


Ilustración 14: Índices de deserción para la carrera “Ingeniería en Informática”

1.2. Estrategias para mitigar la deserción estudiantil en las carreras del Área de Informática

Atento a los datos registrados durante el período 2005-2015, y a la indagación sobre los posibles factores causales de la deserción y el desgranamiento, desde el año 2016 en la UNNOBA se llevan a cabo diferentes estrategias y acciones con la intención de mejorar o revertir esta situación, a saber:

- *Participación en Voluntariados Universitarios.* Se fomenta la participación de estudiantes y docentes en diferentes programas, capacitaciones y actualizaciones. Entre

las más recientes se pueden citar las capacitaciones llevadas a cabo en la Fundación Sadosky, referidas a “la programación y su didáctica” (parte 1 y 2) destinadas a docentes universitarios. En consecuencia esas dos capacitaciones propiciaron la ejecución de tres formaciones a docentes del Nivel Primario y Secundario en esta temática. El objetivo principal de la iniciativa fue impulsar la enseñanza y el aprendizaje de las Ciencias de la Computación en las escuelas argentinas, con énfasis en el aprendizaje por indagación, y procurar despertar la vocación en carreras afines en el Nivel Superior. Particularmente, esta capacitación aborda la estrategia de aprendizaje por indagación utilizando herramientas educativas, lo cual, está íntimamente relacionado con el presente trabajo.

- *Articulación del Nivel Secundario con el Nivel Superior.* Se implementan acciones de manera conjunta orientadas a facilitar el ingreso de los estudiantes a la Universidad y atender sus necesidades de formación, para que puedan transitar con éxito el primer año de los estudios superiores. Con el objetivo de fortalecer los saberes requeridos para asegurar el ingreso a las carreras de la UNNOBA, se dicta los días sábados de 9.00 a 13.00 horas, un Taller semipresencial para los estudiantes del último año del Nivel Secundario, el cual tiene lugar entre los meses de Agosto y Noviembre, y donde se cursan dos asignaturas según la carrera que se elija al finalizar la educación formal¹³. Este taller ofrece una aproximación al universo propio del Nivel Superior: materiales de estudio, metodología de trabajo y de evaluación, cuerpo docente, recursos, infraestructura, entre otros.

Dicha propuesta se divide en módulos que apuntan a trabajar el bagaje que el estudiante trae consigo, así como también incorporar nuevos conceptos y habilidades. En ese sentido, en cada carrera se articula con alguna de las asignaturas que el

¹³ Recuperado de ingreso.unnoba.edu.ar/ingreso

estudiante tendrá durante el primer cuatrimestre de su año inicial. Para el caso de los ingresantes a las carreras del área de informática la articulación se realiza con la asignatura Introducción a la Programación Imperativa, y se dicta el taller denominado “Resolución de Problemáticas en Informática”.

Otra estrategia concerniente al trabajo de investigación y, pensando en que en “la articulación” subyace el concepto de superar compartimentos estancos en educación, de manera que el sistema educativo mejore su calidad, en un marco de diversidad y equidad; es la “*Muestra interactiva del saber y la tecnología*”, que persigue la finalidad de acercar la labor universitaria a los estudiantes del nivel secundario. En esta línea, la ET presenta algunos espacios en los que se visibilizan las ofertas académicas. En el área de informática se ha presentado la actividad “Carrera de programadores”, en la que se invita a los participantes a programar en forma lúdica y con herramientas introductorias a la temática.

- “*Programa de Apoyo y Fortalecimiento Académico*” (PAFA). Es creado por la ET y se orienta a diagnosticar y analizar las características de la matrícula para trazar estrategias que permitan mejorar su acceso y permanencia. El PAFA persigue el objetivo de implementar un sistema integral de asistencia a los estudiantes dividido en tres ejes: tutorías para estudiantes de los primeros años (TUPA), cursos de apoyo para rendir finales de las Asignaturas de los primeros años de las Carreras de la Escuela (CARF) y estrategias de apoyo para la graduación de los estudiantes de informática. En el eje TUPA, que particularmente atañe al presente trabajo, se busca analizar la problemática de los primeros años de estudios universitarios, con el objetivo de lograr un acercamiento social con los estudiantes, donde se ofrezcan experiencias que enriquezcan el vínculo de éstos con la institución. De este modo, se pretende mejorar la permanencia y retención estudiantil en las carreras de la ET, al mismo tiempo que se

investigan los factores por los cuales los estudiantes universitarios desertan en el primero y segundo año.

Si bien los esfuerzos coordinados aún no han alcanzado los resultados esperados, reflejados en el porcentaje de “regularizados” deseados por la Institución, la aplicación de estas estrategias reveló cierta mejoría en la retención, en comparación con los años anteriores.

CAPITULO N°2. DESCRIPCIÓN DE LA ASIGNATURA

Los estudiantes de primer año de las carreras del área de Informática, reciben su formación en los fundamentos de la programación de computadoras a través de dos asignaturas: *Introducción a la Programación Imperativa*, que se cursa en el primer cuatrimestre, y su correlativa *Programación Imperativa* perteneciente al segundo cuatrimestre.

En líneas generales IPI aborda tipos de datos, estructuras de control, y la definición y uso de algoritmos para la resolución de problemas; mientras que en PI se trabajan los conocimientos necesarios para resolver problemas de índole general usando un lenguaje de programación estructurado. En esta última se recuperan todos los conceptos trabajados en la cátedra de IPI, puntualmente: legibilidad, mantenibilidad, modificabilidad y adaptabilidad del código, la comprensión y descomposición de problemas, el desarrollo de Algoritmos, pensados como soluciones a problemas, la modularización y parametrización, la programación estructurada, las estructuras de control y los tipos de datos predefinidos.

3.2 Introducción a la Programación Imperativa

La asignatura IPI forma parte de cuatro planes de estudio de las carreras: Ingeniería en Informática, Licenciatura en Sistemas, Analista de Sistemas y Tecnicatura Universitaria en Soporte Informático, pertenecientes al área de Informática de la ET. En todas estas carreras, con excepción de la Tecnicatura Universitaria en Soporte Informático, la materia se dicta en el primer cuatrimestre del primer año.

Introducción a la Programación Imperativa es una asignatura con un régimen de cursada cuatrimestral y una carga horaria de 96 horas. Sus objetivos principales son

introducir los conceptos básicos de la programación imperativa, resolver problemas mediante un algoritmo, modularizar y parametrizar y aplicar el paradigma estructurado en la resolución de problemas y correctas estructuras de control.

Con respecto a sus contenidos, éstos se distribuyen en cuatro unidades temáticas más una unidad introductoria, a saber:

Unidad Introdutoria: Lógica. Proposiciones. Simbología. Conectores lógicos. Tablas de verdad. Algoritmos. Definición. Sentencias. Pseudocódigo. Estructuras de control.

Unidad 1: El Lenguaje de Programación. Conceptos de intérprete y compilador. Tipos de datos. Tipo de dato bool Variables. Cadena de caracteres. Manejo de las cadenas. Operaciones. Entrada/salida.

Unidad 2: Estructuras de control. Secuencia. Selección. Iteración Lógica en el contexto de Estructuras de Control. Asignación. Comparaciones. Flujo de Control. Ejemplos de uso.

Unidad 3: Por qué modularizar? Función. Invocación Parámetros. Módulos Variables locales a una función. Ámbito global y local Documentación. Precondiciones y Postcondiciones. Manejo de excepciones.

Unidad 4: Estructuras que permiten contener otros objetos. Definición y operaciones. Ejemplos utilizando: tuplas, listas, conjuntos, diccionarios.

Respecto a su encuadre metodológico, en la asignatura se alternan clases teóricas y clases prácticas. Para las primeras, los docentes presentan los temas desde un enfoque conceptual, procurando recrear en los estudiantes los patrones de razonamiento necesarios para la resolución de problemas por medio de algoritmos. Cada clase contempla el uso de diferentes recursos que permiten abordar los contenidos

planificados, como así también la incorporación de nuevas herramientas interactivas que fomentan el dinamismo y la participación del estudiante.

En la instancia práctica las docentes a cargo realizan la transposición didáctica de los conceptos previamente trabajados. Con tal finalidad se resuelven ejercicios en base a problemas concretos, cuyo abordaje puede realizarse de manera grupal, donde la computadora será la herramienta para todo el grupo clase; en parejas, estimulando la confrontación de puntos de vista; de forma individual, reforzando conceptos concretos; o en dinámicas de grupo, para incentivar la integración de los estudiantes.

En líneas generales, se pretende con la cátedra propiciar el análisis de ejercicios prácticos seguido de la confrontación de puntos de vista y diferentes soluciones que pudieran favorecer el enriquecimiento individual del estudiante, valiéndose del trabajo colaborativo.

3.3 Debilidades en el proceso de enseñanza y aprendizaje de la materia IPI

Tal como expresa Steve Jobs: "todo el mundo debería aprender a programar, porque enseña a pensar". Este es un mensaje muy escuchado entre educadores; y se advierte que adquirir habilidades y conceptos relacionados con la programación de computadoras beneficia notablemente al sujeto más allá de pretender o no dedicarse al área.

Sin embargo, la enseñanza y el aprendizaje de la disciplina resulta una actividad intelectual compleja y dificultosa no solo para los estudiantes, sino también para quienes tienen a cargo la enseñanza, sobre todo cuando su impacto es muy importante en la mayoría de las asignaturas sucesivas y en el campo profesional del futuro egresado.

Si bien el dominio de los fundamentos de programación de computadoras es una habilidad esencial a ser desarrollada por los alumnos, los datos estadísticos sobre los resultados de las asignaturas afines a la programación de computadoras y los informes y tutorías de los ingresantes a las carreras del área de informática, revelan una acusada dificultad en la introducción de los estudiantes a la programación imperativa, reflejado en el bajo rendimiento académico, un alto índice de deserción y un significativo desgranamiento en los años posteriores.

A continuación se presentan los datos de asistencia y regularización en la asignatura *Introducción a la Programación Imperativa* entre los años 2010 y 2016:

Año	Asistentes	Regularizados sobre los inscriptos	Regularizados sobre los asistentes
2016	42,55%	20,57%	48,33%
2015	54,35%	18,12%	33,33%
2014	54,41%	30,15%	55,41%
2013	52,76%	24,84%	38,10%
2012	52,76%	29,15%	55,24%
2011	67,00%	33,00%	48,59%
2010	60,00%	28,00%	47,06%

Tabla 2: Datos de la asignatura "Introducción a la Programación Imperativa" en la UNNOBA.

Como se observa en la columna "Asistentes" se discrimina el porcentaje de estudiantes que asistieron a clases, mientras que la columna "Regularizados sobre los inscriptos" los porcentajes de aquellos que se han inscripto, logrando regularizar IPI. Por último, la columna "Regularizados sobre los asistentes" refleja los porcentajes de aquellos que han asistido a clase y lograron regularizar.

Tras un ligero análisis de los datos se puede observar que el porcentaje de los asistentes a clase ha disminuido año tras año. Particularmente, entre 2010 y 2012 los estudiantes regularizados oscilan entre el 28 y el 33 por ciento.

Este contexto hace evidente la importancia de una inminente intervención a través de una acción educativa innovadora y un esfuerzo del equipo docente de cada cátedra para estar atento a las necesidades del educando, tratando de favorecer su proceso de permanencia y promoción hacia años ulteriores.

La inferencia anterior se refuerza si se comparan las últimas dos columnas, en las que se puede observar a simple vista la brecha entre quienes logran regularizar la asignatura asistiendo a clase y quienes no alcanzan dicho objetivo. A partir de allí se podría deducir que la intervención docente sería clave en cuanto a los resultados finales de IPI.

3.4 Estrategias implementadas desde la asignatura Introducción a la Programación Imperativa

Con el objetivo de incrementar el porcentaje de estudiantes regularizados en la asignatura, se han implementado desde el año 2013 una multiplicidad de estrategias que involucran, entre otras cuestiones, la incorporación de cambios en el lenguaje de programación utilizado para abordar los conceptos de la asignatura, la reconfiguración de las dinámicas de las clases teóricas y prácticas, la capacitación del plantel docente en didáctica de la programación imperativa y la organización de encuentros presenciales y virtuales para facilitar el intercambio dentro del equipo.

En lo que respecta al año 2017, teniendo en cuenta que IPI es una asignatura troncal del primer año de las carreras del área de informática, se incorporó a las estrategias vigentes el siguiente esquema de trabajo.

Se propuso el dictado de clases teóricas y prácticas, apoyadas en el Entorno Virtual de Enseñanza y Aprendizaje (EVEA) para poner a disposición de los estudiantes el material teórico y actividades planificadas para la cursada, como así también el reglamento de la asignatura.

En la siguiente ilustración se puede observar una captura de pantalla del curso para la asignatura IPI en el EVEA “UNNOBA Virtual”:

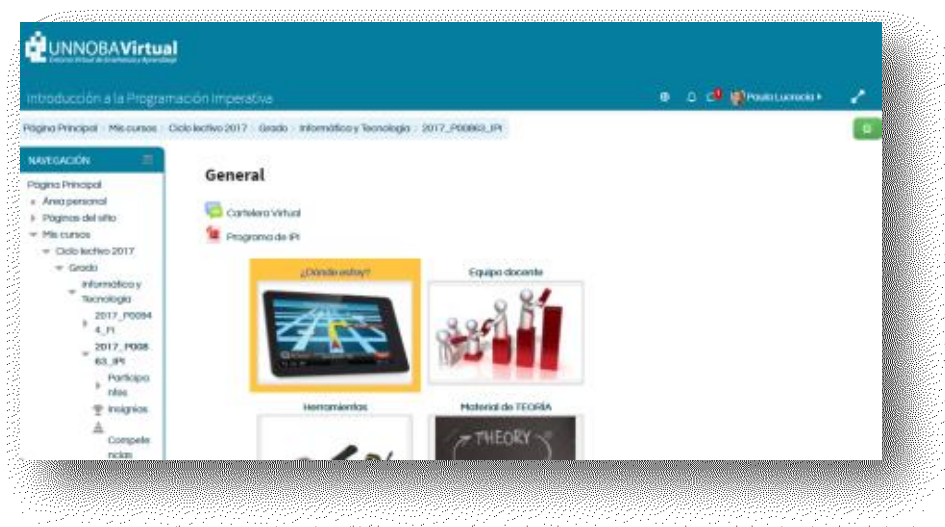


Ilustración 15: curso para IPI en UNNOBA VIRTUAL¹⁴

Clases teóricas.

En las clases teóricas se presentaron las unidades temáticas desde un enfoque conceptual, procurando recrear en los estudiantes patrones de razonamiento adecuados para la resolución de problemas por medio de algoritmos.

A continuación, se muestran capturas de pantalla del curso en el EVEA con las presentaciones utilizadas en las clases teóricas, así como sus respectivos documentos PDF para facilitar la impresión del material (observar las ilustraciones 16 y 17).

¹⁴ virtual.unnoba.edu.ar

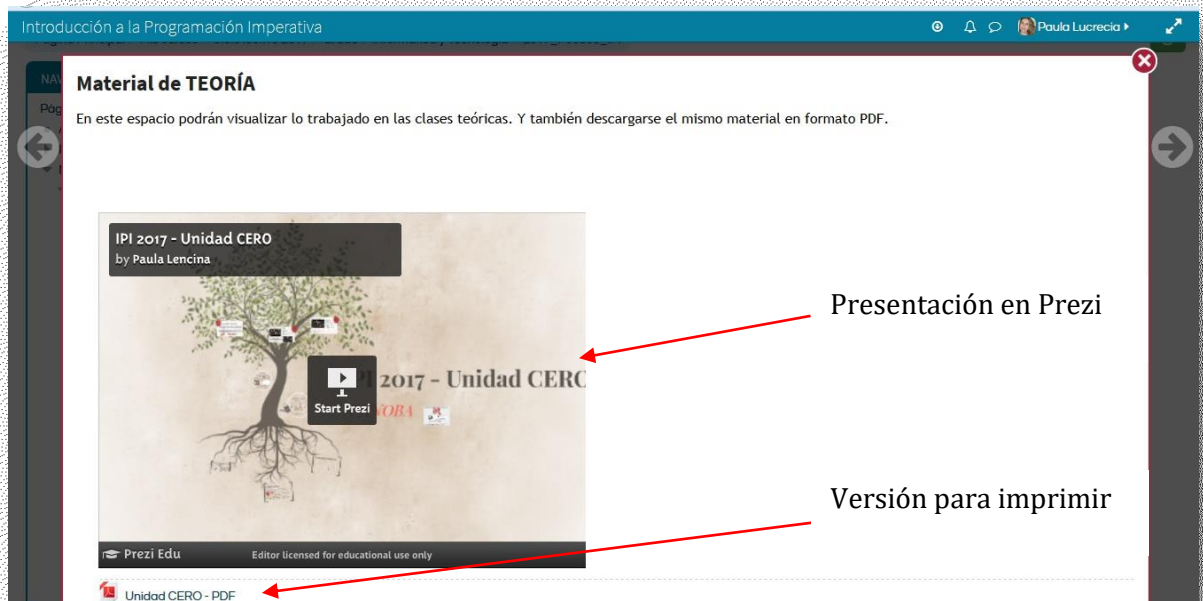


Ilustración 16: captura de pantalla del curso en el EVEA

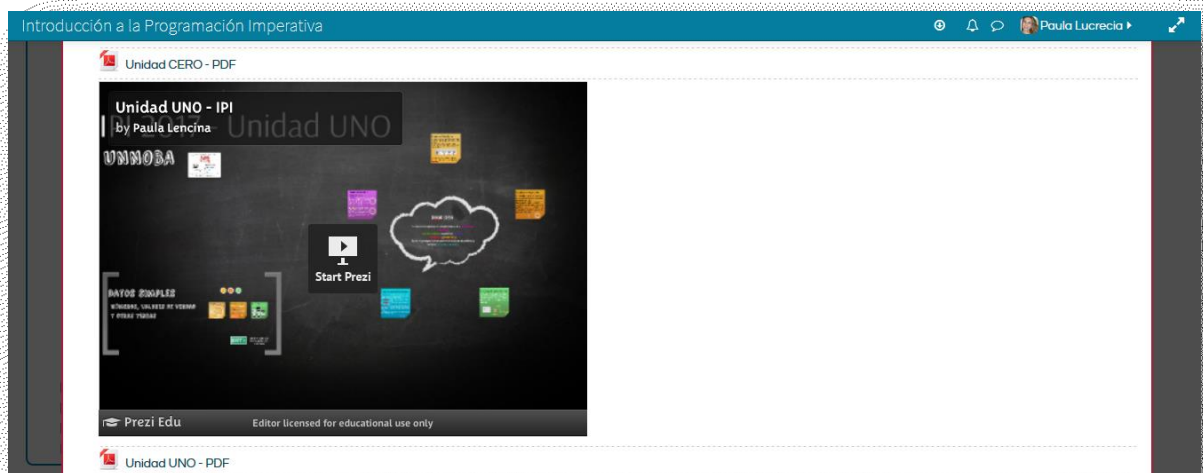


Ilustración 17: captura de pantalla del curso en el EVEA

En IPI se reforzaron los distintos conceptos por medio de ejemplos prácticos con un grado de complejidad creciente conforme transcurría el cuatrimestre. Esto se hizo con el uso de dos herramientas: un pseudocódigo propio y un lenguaje de programación.

Es importante aclarar que, por resolución de la universidad, en IPI y en el resto de las asignaturas correspondientes al primer cuatrimestre de todos los primeros años, se añadió un mes de trabajo introductorio. En ese contexto se trabajó con un pseudocódigo construido en el marco de la formalización de material diseñado ad hoc.

El mundo de Ada.

A continuación, se exponen algunas imágenes del material antes mencionado.



Ilustración 18

En la Ilustración 18 se puede visualizar una captura de pantalla del material disponible en el EVEA con el pseudocódigo cuyo personaje es llamado Ada.¹⁵

¹⁵ El personaje de Ada, protagonista de este pseudocódigo, está inspirado en la célebre Ada Lovelace, mejor conocida como la “primera programadora de la historia”. Su nombre real era Ada Augusta Byron King nacida el 10 de diciembre de 1815 en Inglaterra y fallecida el 27 de noviembre de 1852, con tan sólo 36 años de edad.

Ada amaba las matemáticas y las ciencias. Conoció a Charles Babbage, matemático, ingeniero e inventor británico, quien diseñó las máquinas calculadoras programables. Ambos quedan admirados el uno con el otro y comenzaron a trabajar juntos. De este modo Ada se inicia en el fascinante mundo de la programación.

Luego de comenzar con conceptos elementales como algoritmos y programas, se presentaba “El mundo de Ada”, lo que permitía trabajar con la herramienta del pseudocódigo y sus diferencias con el uso de un lenguaje de programación.

En la ilustración 19 se puede observar la ciudad de Ada, compuesta por nueve (9) manzanas y nueve (9) calles, enumeradas de uno (1) a nueve (9).

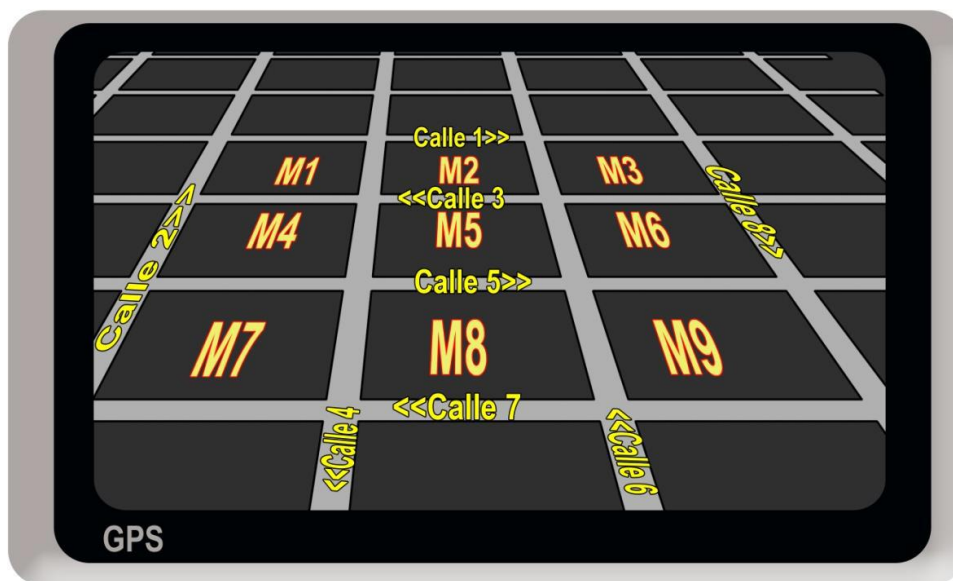


Ilustración 19: Ciudad por la que circula Ada.

El pseudocódigo incluía las siguientes sentencias:

- Caminar una cuadra
- Doblar hacia la derecha
- Doblar hacia la izquierda
- Recoger un residuo
- Colocar en el cesto de residuos
- Cruzar la calle
- Esperar
- Informan cantidad de.....

Además, las siguientes condiciones:

El recorrido comienza siempre en calle 1 y calle 2, mirando para calle 4,
Por las calles pasan autos,
Hay árboles en las esquinas,
Hay residuos en las esquinas,
Hay un local de venta de repuestos,
Hay un local de venta de electrodomésticos,

Preguntas que pueden realizarse para evaluar condiciones:

- ¿Pasan autos?
- ¿Hay árboles?
- ¿Hay residuos?

Y las estructuras de control como se describen a continuación:

- Si
- Mientras (condición o condiciones)
- Repetir X-veces

Se presentaban enunciados tales como: indicarle a Ada que recorra el perímetro del cuadrado que se forma al recorrer las manzanas: M1, M2, M5 y M4, como se observa en la Ilustración 20; por las calles: 1, 6, 5 y 2.

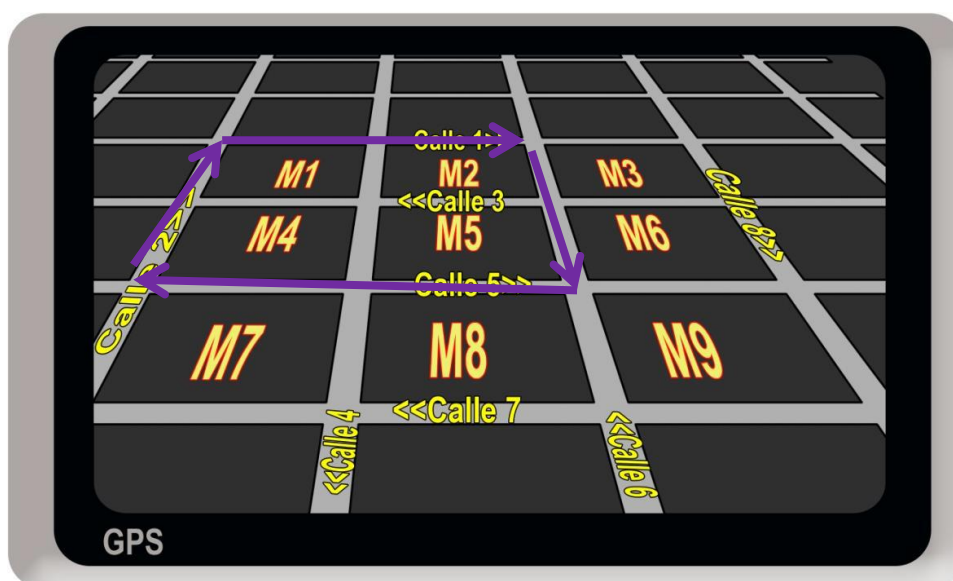


Ilustración 20

Una solución planteada fue:

```
ALGORITMO
  Caminar una cuadra
  Cruzar la calle
  Caminar una cuadra
  Doblar a la derecha

  Caminar una cuadra
  Cruzar la calle
  Caminar una cuadra
  Doblar a la derecha

  Caminar una cuadra
  Cruzar la calle
  Caminar una cuadra
  Doblar a la derecha

  Caminar una cuadra
  Cruzar la calle
  Caminar una cuadra
  Doblar a la derecha
```

La solución propició el planteo de dos preguntas que llevaron a la introducción de otros conceptos. Por ejemplo: ¿hay algo que se repite una y otra vez? ¿Se podrán identificar patrones dentro del algoritmo planteado? Esas cuestiones llevaron a los estudiantes a advertir la presencia de patrones dado que las líneas:

```
  Caminar una cuadra
  Cruzar la calle
  Caminar una cuadra
  Doblar a la derecha
```

se reiteraban cuatro (4) veces. Esto dio lugar al uso de la estructura de control “Repetir X-veces”, entonces el algoritmo pudo escribirse del siguiente modo:

```
ALGORITMO
  Repetir 4 veces
    Caminar una cuadra
    Cruzar la calle
    Caminar una cuadra
    Doblar a la derecha
```

A partir de la solución anterior surgieron interrogantes como ¿qué sucede si pasan autos a la hora de cruzar? En ese caso habrá que preguntar algo que permita corroborar que Ada cruce sin peligro. Una solución posible podría ser:

```
ALGORITMO
  Caminar una cuadra
  Mientras ¿Pasan autos?
    Esperar
  Cruzar la calle
  Caminar una cuadra
  Doblar a la derecha

  Caminar una cuadra
  Mientras ¿Pasan autos?
    Esperar
  Cruzar la calle
  Caminar una cuadra
  Doblar a la derecha

  Caminar una cuadra
  Mientras ¿Pasan autos?
    Esperar
  Cruzar la calle
  Caminar una cuadra
  Doblar a la derecha

  Caminar una cuadra
  Mientras ¿Pasan autos?
    Esperar
  Cruzar la calle
  Caminar una cuadra
  Doblar a la derecha
```

Otra solución planteada teniendo en cuenta la mejora en el segundo algoritmo fue:

```
ALGORITMO
  Repetir 4 veces
    Caminar una cuadra
    Mientras ¿Pasan autos?
      Esperar
    Cruzar la calle
    Caminar una cuadra
    Doblar a la derecha
```

De este modo se trabajó con el Mundo de Ada, durante el primer mes de la cursada de IPI. La experiencia y el material utilizado para ese año de cursada

propiciaron el desarrollo de un material de tipo Libro denominado “Resolviendo problemas de informática”¹⁶.

Trabajando con Python.

La segunda herramienta trabajada en IPI fue el intérprete Python que permitió plantear ejercicios de programación mediante la ejecución de código con muestra inmediata de resultados.

Python es uno de los lenguajes de programación dinámicos más populares que existen. Es un lenguaje interpretado, dado que el código fuente de un programa escrito en Python es traducido por un intérprete, que realiza las acciones indicadas en cada una de las instrucciones leídas mientras lo está ejecutando.

Para poder correr un programa escrito en un lenguaje interpretado es necesario disponer del código fuente del programa y además tener instalado un intérprete para ese lenguaje.

En su sitio web python.org se lo describe como “Poderoso... y rápido”, que permite “jugar” con las demás herramientas, “corre en todos lados”, es amigable, fácil de aprender y es abierto.

En el sitio se invita a participar de la comunidad de Python, una comunidad considerable cuyo objetivo de crecer. “Python’s community” tiene como base apoyar al programador principiante, invitando a los expertos a sumarse en el enriquecimiento de la base de conocimiento de código abierto.

¹⁶ ISBN 978-987-3832-48-2. Ciudad Autónoma de Buenos Aires. Alfaomega Grupo Editor Argentino. 2019

Es un lenguaje multiplataforma, que posee una licencia de código abierto, denominada “Python Software Foundation License” y se puede descargar de su sitio oficial¹⁷.

Si bien es un lenguaje orientado a objetos, soporta estilos de programación procedural y funcional, lo que fundamenta su elección para introducir a los estudiantes de la UNNOBA en la programación imperativa.

Teniendo en cuenta que, a diferencia de otros lenguajes como Java o C, presenta una sintaxis sencilla, utiliza tipos dinámicos, administra automáticamente la memoria a través de recolección de basura, incluye múltiples bibliotecas de clases, y que la comunidad educativa lo incorpora desde hace tiempo como herramienta válida para los inicios de la programación, se lo valora como un lenguaje propicio en la UNNOBA.

A pesar de las virtudes de ese lenguaje en materia de enseñanza en la programación imperativa, se ha observado que aún persisten dificultades para el aprendizaje directo en un lenguaje de programación, cuando este es la primera herramienta de acercamiento del alumno. En este sentido se considera el uso de un pseudocódigo para un primer acercamiento, considerando que lo importante es que el estudiante aprenda a programar más allá de la herramienta. Y la simplicidad que proporciona el pseudocódigo (como el de la interfaz del REP) es una opción a considerar.

En las ilustraciones 21 y 22 se puede apreciar parte del material diseñado para IPI con ejemplos en pseudocódigo y Python.

¹⁷ python.org

Si la condición tiene más de dos respuestas?

```

>>> instrucción 1
>>> if condición 1 es True:
    instrucción 2
else:
    if condición 2 es True:
        instrucción 3
    else:
        if condición 3 es True:
            instrucción 4
        else:
            instrucción 5
>>> instrucción 6
  
```

puede resultar ilegible

Ilustración 21

for/in

Itera sobre los **elementos** de cualquier **secuencia**, en el **orden** determinado por la misma secuencia.

La estructura "for/in" se puede usar para iterar sobre los valores numéricos generados por "range", sobre los caracteres de una **cadena de caracteres** y sobre los elementos de una **tupla**.

Por ejemplo:

```

>>> for x in range(10)
... print(x)

>>> for x in 'Una cadena':
... print(x)

>>> for x in (1, 2, 3):
... print(x)
  
```

Break y Continue

Ilustración 22

Clases prácticas.

Las clases prácticas, llevadas a cabo en los laboratorios de informática de la universidad, comenzaban con una breve explicación de los ejes teóricos para recuperar, reforzar y articular lo abordado desde la teoría.

Se intentó en cada encuentro propiciar el análisis de ejercicios prácticos seguido de la confrontación de puntos de vista y diferentes soluciones que pudieran favorecer el enriquecimiento individual del estudiante, valiéndose del trabajo colaborativo.

3.4.1 Herramientas disponibles para la enseñanza de la programación en la UNNOBA

Tal como se ha expresado en secciones preliminares, el extenso recorrido profesional de la autora, en el rol de docente la UNNOBA en las asignaturas IPI y PI, ha permitido concretar el estudio en el tiempo previsto y con los recursos requeridos para su abordaje.

Asimismo, su desempeño como capacitadora en “la programación y su didáctica” (parte 1 y 2), facilitó el acercamiento a diferentes herramientas para mejorar la enseñanza de la programación y aplicarlas durante las capacitaciones brindadas a docentes de nivel primario y secundario, como así también en diferentes propuestas educativas, como cursos en Extensión Universitaria en la UNNOBA destinada a niños y adolescentes.

A continuación se describen las herramientas disponibles para la enseñanza de la programación. Cabe aclarar que si bien algunas son utilizadas en la UNNOBA, éstas no se encuentran enmarcadas en la propuesta educativa de IPI.

Actividades desenchufadas.

Su nombre alude a que para ejecutar los algoritmos no se requiere de la tecnología, como por ejemplo de un lenguaje de programación, sino que solo se necesita del cuerpo y de otros materiales, como pizarra y fibrones.

A continuación se describe una actividad realizada en los Cursos de Extensión universitaria de la UNOBA, dirigida a niños y adolescentes “Programá y creá tu video juego”. Una actividad similar se ha llevado a cabo al comienzo de la capacitación dictada sobre la didáctica de programación, dirigida a docentes de nivel primario y secundario.

La actividad denominada “Sacar a la docente del aula”, persigue dos objetivos complementarios: fomentar el trabajo en equipo, en términos de comunicación, cooperación y empatía, y favorecer el desarrollo de la autoconfianza de los estudiantes, ya que no solo se presenta como una actividad de fácil resolución, sino que también propicia una atmósfera desestructurada y placentera para el aprendizaje.

Descripción de la actividad. La actividad se divide en dos partes. En una primera etapa la docente se coloca en un punto estratégico y solicita a los estudiantes que, de manera conjunta, le indiquen los pasos a seguir para salir del aula. Para animar órdenes concretas y evitar todo tipo de ambigüedad en las instrucciones, la docente exagerará algunos movimientos. Por ejemplo si se le indica “caminar” podría hacerlo sin frenar, tal vez chocándose algún obstáculo. De igual modo si la orden fuera “girar” puede que lo haga en un sentido que no sea el solicitado o bien que gire 360 °. Seguidamente, la docente se coloca estratégicamente en otro lugar del aula, o en otra posición o con diferentes obstáculos, y solicita al grupo clase que indique los pasos a seguir para salir del aula. A modo de cierre, y en una segunda etapa, la docente podrá abordar los conceptos que desea, recuperando ejemplos ilustrativos de la actividad.

Otra actividad fue “El baile del Fornite”, llevada a cabo durante el festejo de la primavera 2018. En las fotos que se presentan a continuación, puede observarse a niños entre ocho (8) y doce (12) años de edad, diseñando un algoritmo para indicarle a la docente cómo realizar “El baile del Fornite”.



Ilustración 23: niños trabajando en equipo en el armado de un algoritmo.

La actividad consistía en agruparse en equipos de trabajo y consensuar la manera de armar un algoritmo a partir de ciertas instrucciones, eventos y estructuras de control que la docente había escrito anteriormente en cartulinas con formas que encastraban estratégicamente. Esta actividad propició la comunicación constante entre los compañeros de equipo, para analizar el problema planteado y proponer una solución consensuada.



Ilustración 24: alumnos del curso trabajando al aire libre en uno de los edificios de la UNNOBA.

Luego le presentaban el algoritmo a la docente, quien siguiendo las instrucciones, intentaba bailar.

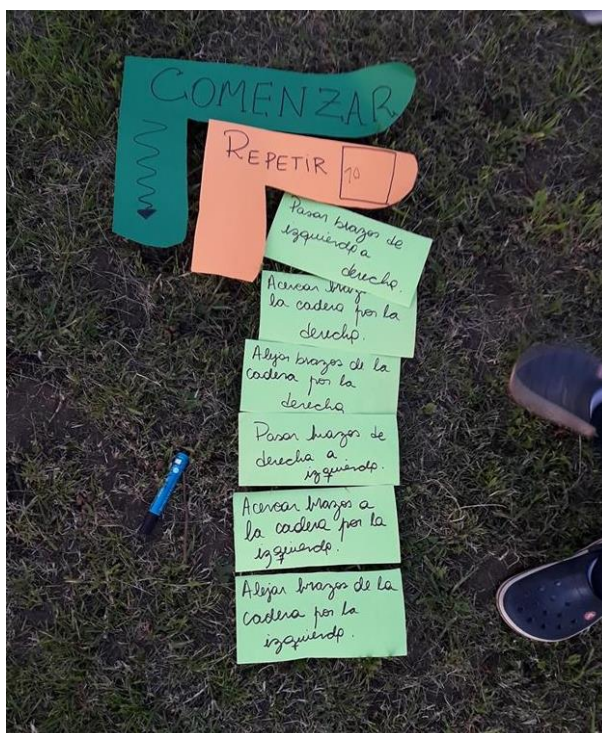


Ilustración 25: algoritmo armado por un grupo

Actividades “Desenchufadas” como las anteriormente descritas, pueden pensarse y adaptarse a lo que se desee trabajar, básicamente porque, tal como su nombre lo indica, no se limitan ni dependen de ninguna tecnología de la información y la comunicación.

Pilas Bloques.

Esta herramienta también fue empleada para abordar conceptos de programación en los Cursos de Extensión universitaria de la UNNOBA destinados a niños y adolescentes; y en la capacitación sobre la didáctica de programación, dictada para docentes de los niveles primario y secundario.

Pilas bloques, desarrollo que le pertenece al equipo del proyecto Program.AR. de la Fundación Sadosky, es un entorno en el que se plantea el armado de programas por

medio de bloques. Ofrece un lenguaje visual, creado para facilitar el aprendizaje intuitivo, que muestra de una manera animada aquello que se está ejecutando.

Que la herramienta tenga bloques, significa que no será necesario que el programador escriba el código, sino que va a contar con diferentes piezas que deberá poder ensamblar. Como en un rompecabezas, no todas las piezas encajan, por lo que hay que buscar la manera de insertar un bloque con otro. Así será más sencillo armar un programa.

Pilas Bloques plantea pequeños retos utilizando estrategias para la resolución de problemas. Estos desafíos van incorporando conceptos y habilidades a medida que se avanza, y divide los problemas en dos grandes grupos: Desafíos del primer ciclo y Desafíos del segundo ciclo.

Particularmente se destaca que su uso ha resultado sencillo, como así también ha propiciado el abordaje de los conceptos que se querían trabajar.

A continuación se presentan algunas imágenes del sitio web:
<http://pilasbloques.program.ar/>

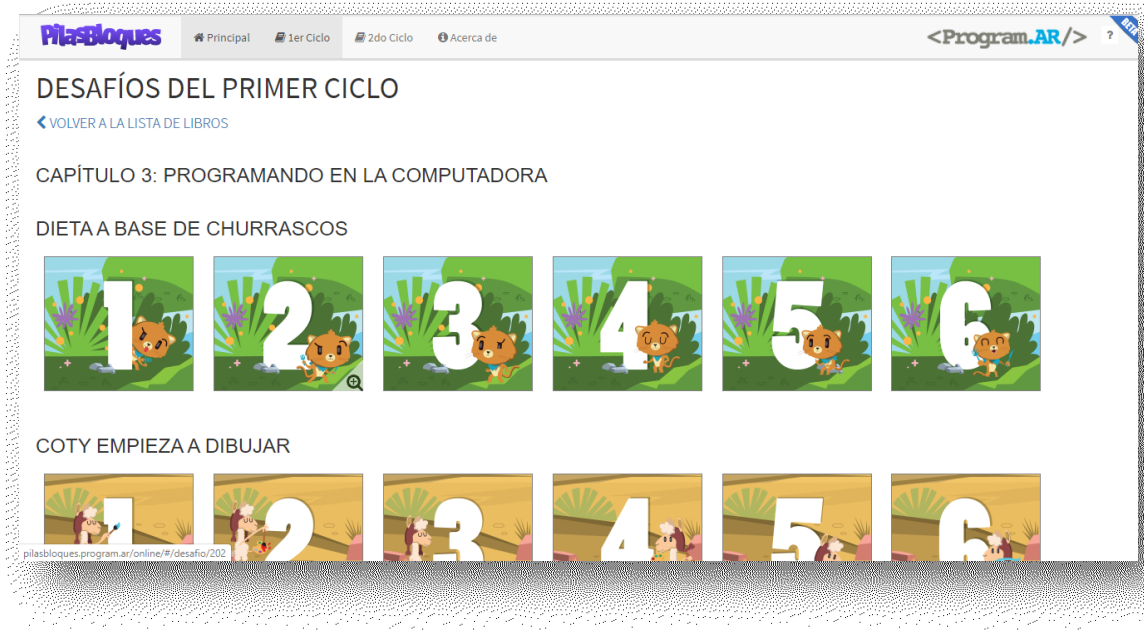


Ilustración 26: captura de la web de Pilas Bloques.

En la Ilustración 26 se pueden apreciar los ejercicios indicados para el primer ciclo escolar.

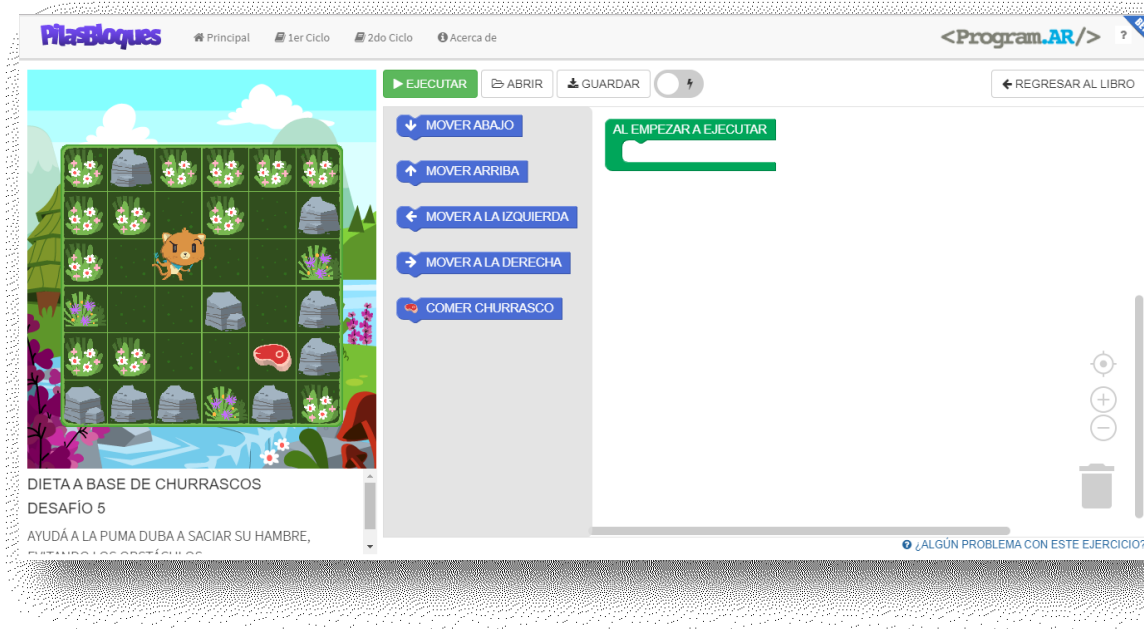


Ilustración 27: uno de los desafíos para el primer ciclo

En la Ilustración 27 se visualiza un ejercicio para el primer ciclo escolar, “Dieta a base de churrascos”.

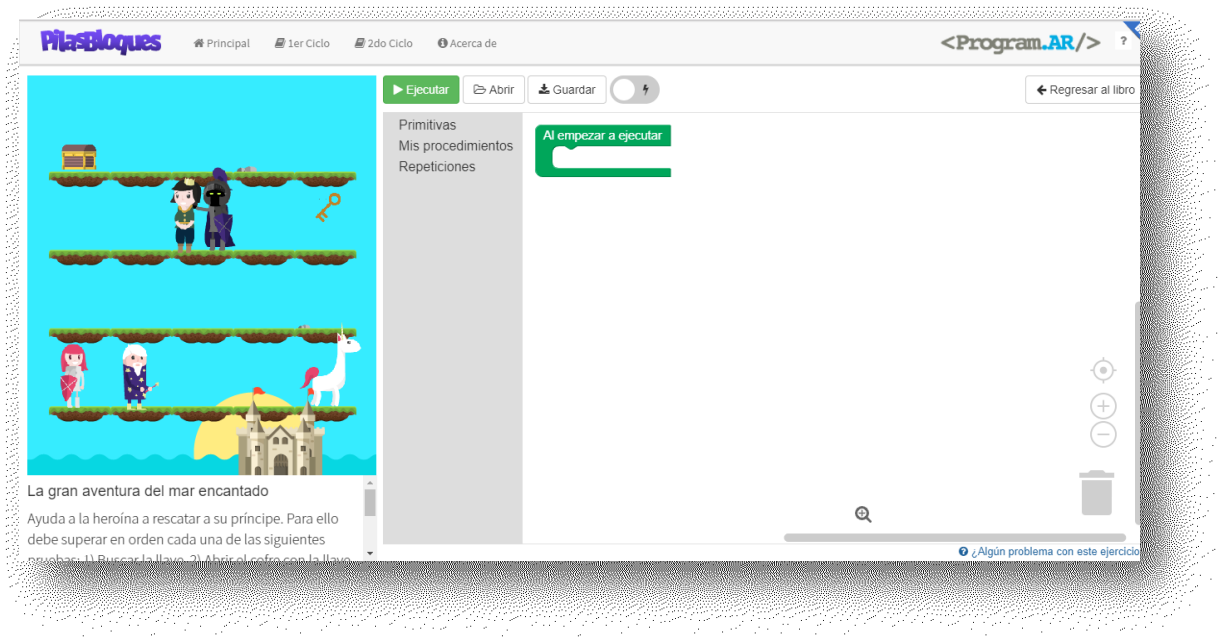


Ilustración 28: uno de los desafíos para el segundo ciclo

En la Ilustración 28 se pueden observar los ejercicios propuestos para el segundo ciclo escolar.

ALICE

Esta herramienta aún no se ha probado en los cursos para niños y adolescentes de la UNNOBA, pero sí en la capacitación dictada sobre la didáctica de programación dirigida a docentes de los niveles primario y secundario.

Alice es un entorno de programación 3D desarrollado en Java. Fue creado por la Carnegie Mellon University en el año 2004, y básicamente sirve para el diseño de juegos y animaciones. Se lo considera como una herramienta intuitiva y de fácil uso, por ello los estudiantes aprenden conceptos de programación orientada a objetos. Se utiliza arrastrando y soltando objetos que luego serán animados, se construyen programas, visualizándolos de forma gráfica.

El uso de Alice permite un acercamiento agradable y atractivo a la programación, presentando conceptos fundamentales de una manera divertida estimulando un aprendizaje visual donde se puede lograr una animación sin errores.

A diferencia de Pilas Bloques, Alice presenta un entorno “vacío” al no disponer de ejercicios prediseñados para completar, sino que expone las herramientas para que el usuario cree libremente (observar la Ilustración 29).

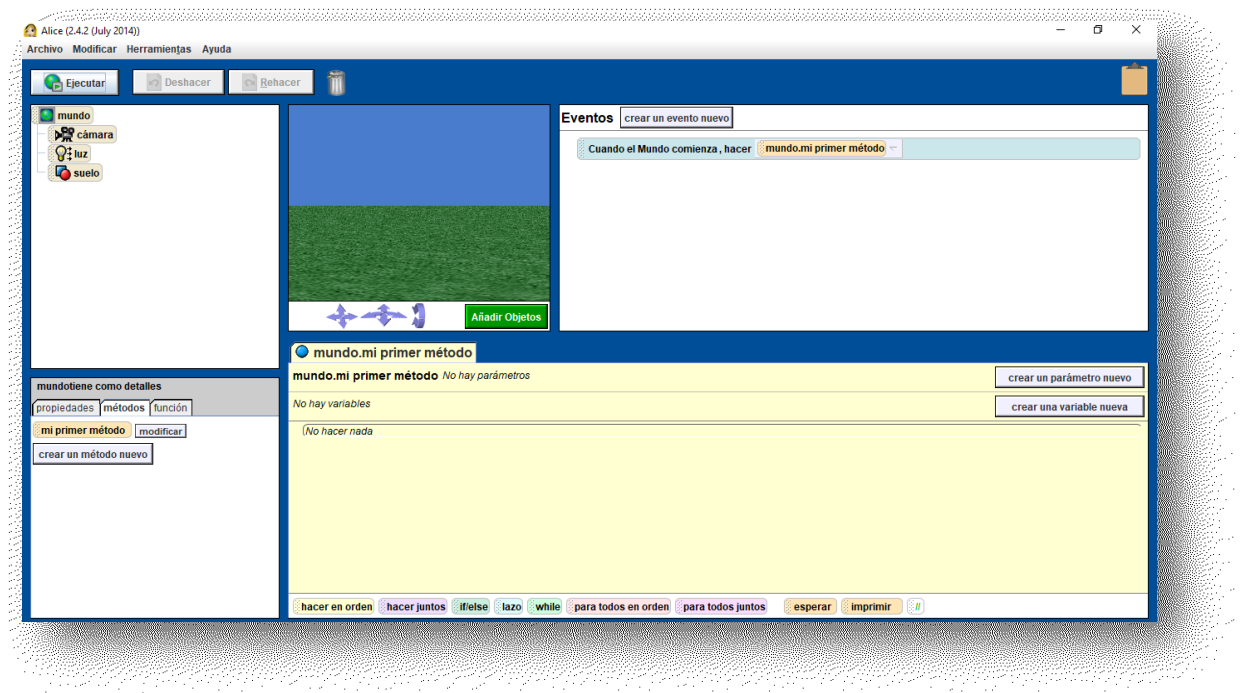


Ilustración 29: interfaz de Alice al abrir la aplicación.

Tal como se describe en su sitio web “Alice es utilizada por maestros en todos los niveles, desde escuelas intermedias (y a veces incluso más jóvenes) hasta universidades, en aulas escolares y en la programación después de la escuela y fuera de ésta, y en materias que van desde artes visuales y artes del lenguaje hasta los fundamentos de la programación y la introducción a los cursos de java.”¹⁸

SCRATCH

¹⁸ <https://www.alice.org/about/>

Scratch fue otra de las herramientas utilizadas para abordar conceptos de programación en los Cursos de Extensión universitaria de la UNOBA y en la capacitación sobre la didáctica de programación.

Al igual que Alice, exhibe las herramientas para que el usuario cree libremente (observar la Ilustración 30).

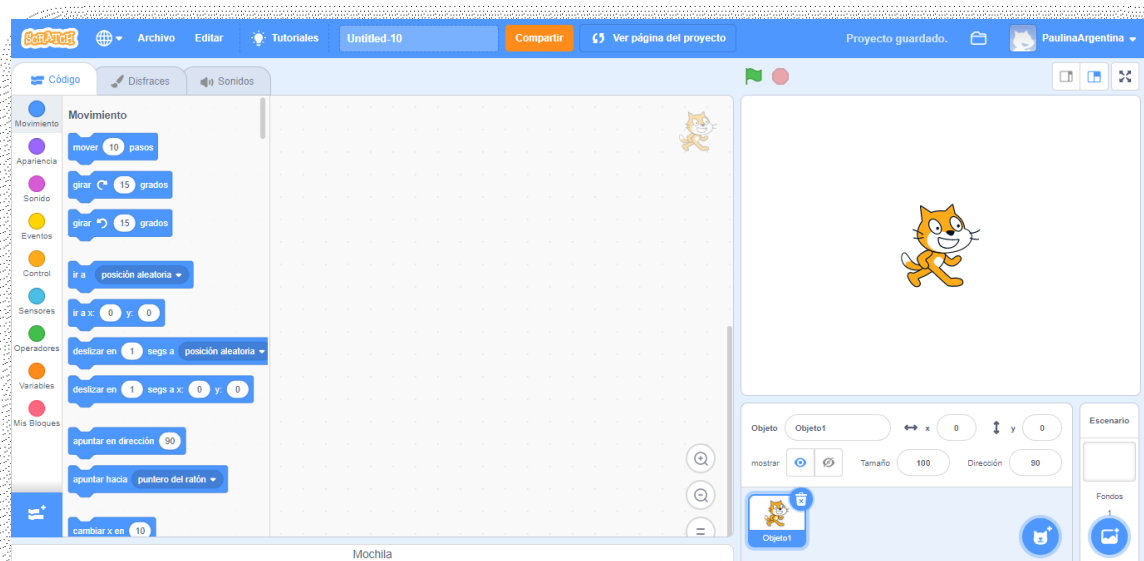


Ilustración 30

En la Ilustración anterior puede observarse que el entorno, al igual que en Pilas Bloques, es un entorno en el que se plantea el armado de programas por medio de bloques.

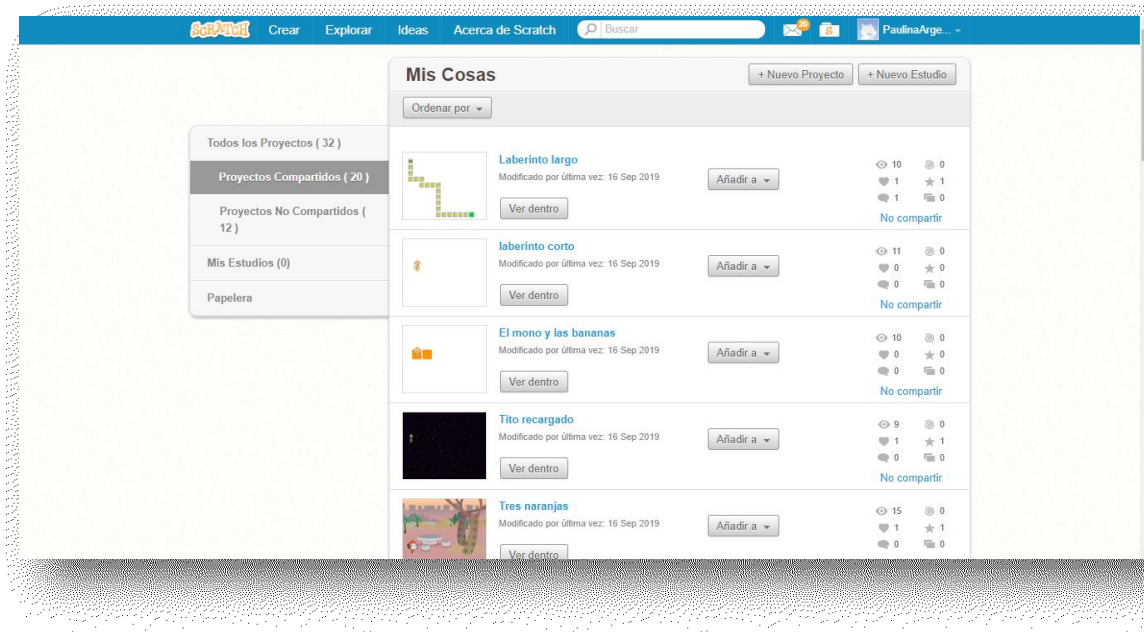


Ilustración 31: proyectos compartidos con la comunidad.

Se puede resumir entonces que los resultados del trabajo con niños y adolescentes jóvenes, han sido motivadores.

GOBSTONES

En la UNNOBA se ha utilizado Gobstones, en el dictado de una capacitación a docentes de los niveles primario y secundario.

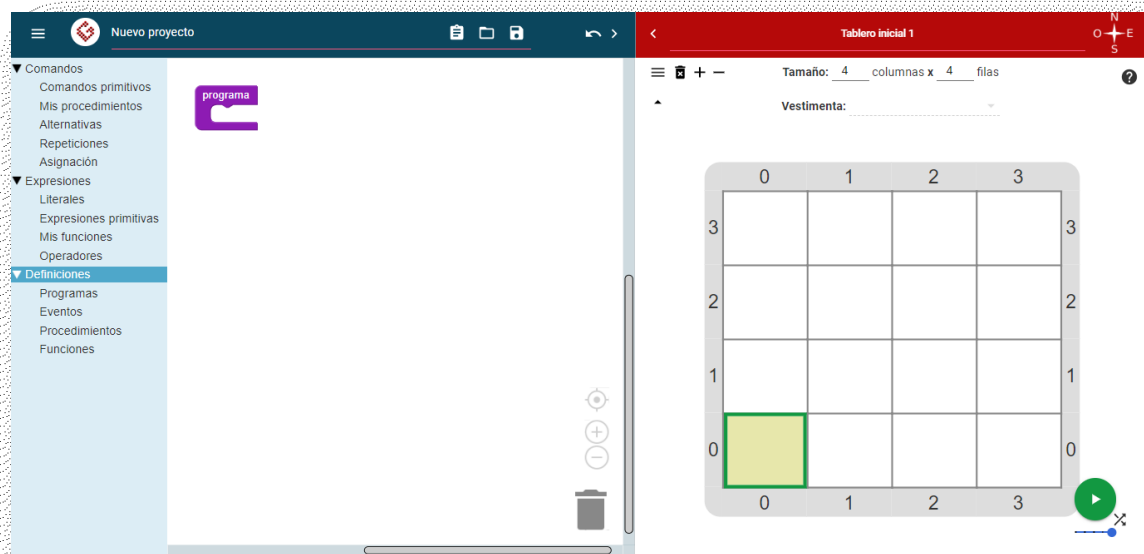


Ilustración 32: captura de pantalla de Gosbtone al inicio.

Gobstones es un lenguaje de programación, cuya autoría le pertenece a la Universidad Nacional de Quilmes (UNQ), diseñado como herramienta para introducir a los estudiantes en los conceptos de la programación.

En la UNNOBA los resultados de la experiencia, la cual se limita al dictado de la capacitación a docentes, ha sido provechosa quizá por la similitud que presenta Gobstones con otras herramientas. Pero aún no se ha incursionado en lo que respecta a la enseñanza a estudiantes universitarios de la UNNOBA.

APPINVENTOR

App Inventor es un entorno de desarrollo de software creado por Google Labs para la elaboración de aplicaciones destinadas al sistema operativo Android. El usuario puede ir enlazando una serie de bloques para crear la aplicación, en un entorno como el que puede apreciarse en la Ilustración 33:

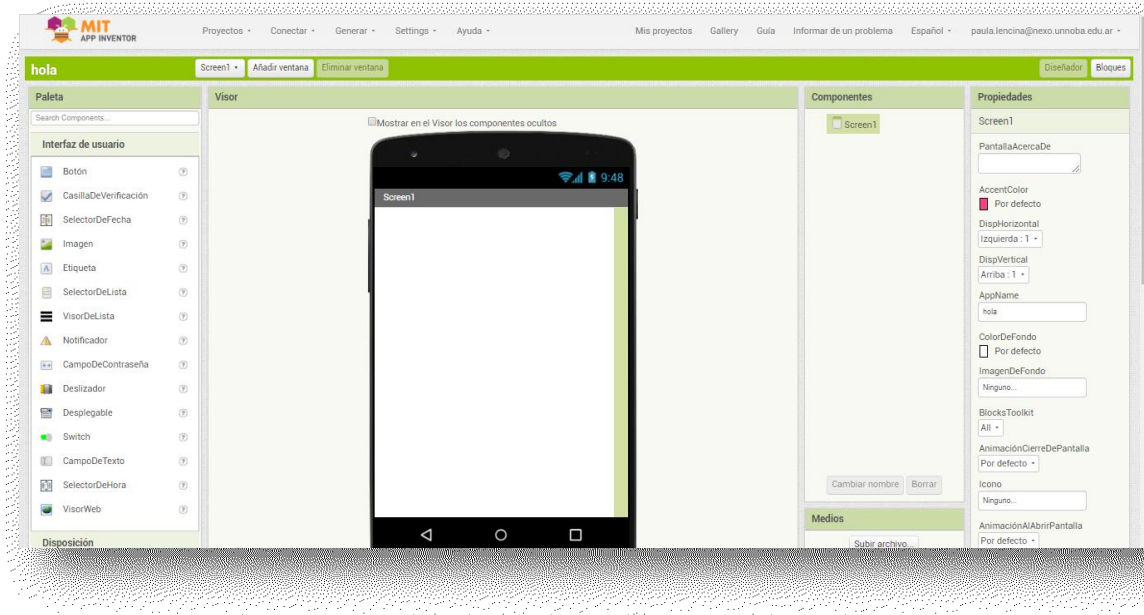


Ilustración 33

Tal como se indica en su sitio web:

“MIT App Inventor es un entorno de programación visual e intuitivo que permite a todos, incluso a los niños, crear aplicaciones totalmente funcionales para teléfonos inteligentes y tabletas. Los nuevos en MIT App Inventor pueden tener una primera aplicación simple en funcionamiento en menos de 30 minutos. Y lo que es más, nuestra herramienta basada en bloques facilita la creación de aplicaciones complejas y de alto impacto en mucho menos tiempo que los entornos de programación tradicionales. El proyecto MIT App Inventor busca democratizar el desarrollo de software al empoderar a todas las personas, especialmente a los jóvenes, para pasar del consumo de tecnología a la creación de tecnología¹⁹.”

En la UNNOBA se ha utilizado en el dictado de una capacitación a docentes de los niveles primario y secundario, pero la experiencia no resultó absolutamente positiva

¹⁹ <https://appinventor.mit.edu/>

dado que, si bien la herramienta es atractiva y motivadora, las problemáticas que se presentaron en relación a la conexión muchas veces dieron la sensación de fracaso en los docentes capacitados.

CAPÍTULO N°3. METODOLOGIA PROPUESTA.

3.5 El Robot Educativo Programable (REP)

A partir de la utilización de la minería de datos se planifica un modelo de intervención áulica virtual que permite diseñar y ejecutar estrategias tecnológicas. En el marco del proyecto “Informática y Tecnologías Emergentes” de la UNNOBA, presentado durante la convocatoria SIB 2017, se ha trabajado en la línea “Robótica e Interacción Hombre-Máquina (HCI)”. Esta línea incluye la investigación, diseño, construcción y programación de robots como solución a problemas multidisciplinarios y transversales. Además, encierra la investigación de metodologías y herramientas que contribuyen a un cambio de paradigma hacia la interacción ubicua, entornos inteligentes y tecnologías interactivas de apoyo, en cualquier aspecto de la vida y las actividades humanas desde una perspectiva global y social.

Dentro del Instituto de Investigación y Transferencia de Tecnológica de la UNNOBA un grupo de investigadores han trabajado en la construcción de un prototipo del Robot Educativo Programable. En la Ilustración 34 puede observarse la imagen del REP.

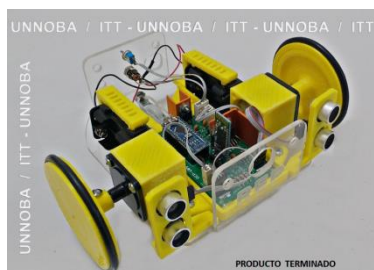


Ilustración 34: Imagen de REP

Esta labor se llevó a cabo a fin de contar con una herramienta propia de la universidad para trabajar a futuro, entre otras cuestiones, en la enseñanza de la

programación de computadoras. Se estima que el uso de robots podría tener un alto potencial de permitir la aplicación más tangible y concreta de las habilidades de programación.

La construcción del prototipo se desarrolló en el marco de un trabajo de investigación denominado “Herramientas didácticas y nuevas formas de interacción para la inclusión de TIC en la enseñanza” cuyo objetivo general fue analizar y desarrollar herramientas y estrategias didáctico-pedagógicas que permitan el diseño y la evaluación de actividades para la inclusión de las TIC en las aulas del Nivel Primario y Secundario, en articulación con el Nivel Superior.

Bajo la línea Robótica e Interacción Hombre-Máquina (HCI), se trabaja entonces en busca de estrategias educativas con respecto al paradigma imperativo de la programación de computadoras, utilizando como herramienta al REP, pero esta vez en la Universidad. Particularmente se estudiará el caso de la asignatura: “Introducción a la Programación Imperativa”.

Con tal finalidad, se diseñó una interfaz para que el REP pudiera emplearse como herramienta para el caso de estudio seleccionado. Se parte de la siguiente hipótesis: la interfaz para el uso del REP permitirá establecer una forma diferente de acercar a los estudiantes a los contenidos, y motivarlos mediante el uso de un robot, siempre y cuando, a través de la intervención del docente, se logre un diseño de actividades motivadoras y estimulantes.

3.6 Desarrollo de una interfaz para el REP

El desarrollo de la interfaz responde a la necesidad de poder usar el REP para abordar en forma didáctica conceptos de programación de computadoras por parte de los estudiantes ingresantes de la universidad.

En informática interfaz es el nombre que se le da a la conexión funcional entre dos sistemas, programas, dispositivos o componentes de cualquier tipo; que proporcionan una comunicación de distintos niveles permitiendo el intercambio de información. También puede describirse como el conjunto de características y elementos que un programa de ordenador presenta en pantalla para facilitar la interacción entre el usuario y el programa.

En ese sentido se deben considerar algunos aspectos fundamentales como por ejemplo que la interfaz debe ser lo más amigable posible a fin de facilitar su uso. Por otra parte se debe considerar qué es lo que se desea que haga el robot (REP) y cuáles serán sus alcances y limitaciones. Dependiendo de esas características específicas se trabajará en el desarrollo de una interfaz que “reciba” sentencias en pseudocódigo, con ayudas en Python, y luego lo transmita al REP para que este pueda ejecutar.

Una vez en claro estas cuestiones, se intentará proporcionar a la interfaz una suficiente cantidad de acciones posibles, a fin de lograr variadas y numerosas actividades que el REP pueda realizar.

3.6.1 Características del funcionamiento de la interfaz

El REP está diseñado con tecnología Arduino²⁰, y la programación de este micro-controlador está basada en C++.

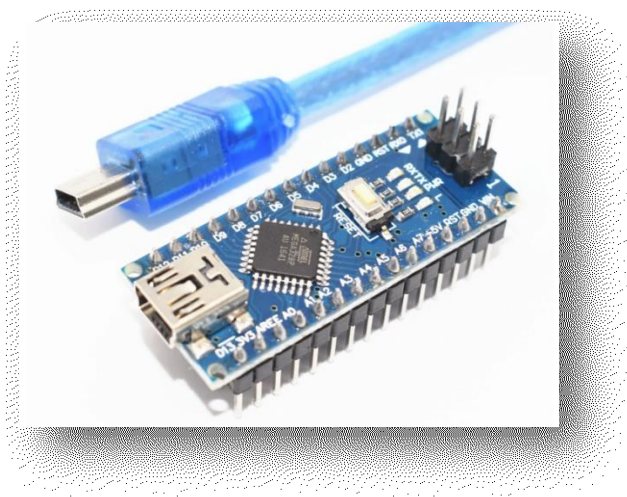


Ilustración 35: imagen de una placa Arduino Nano

En la Ilustración 35 puede observarse una placa Arduino Nano. Ésta es una pequeña y completa placa de desarrollo basada en el ATmega328 (Arduino Nano 3.0) o el ATmega168 en sus versiones anteriores (Arduino Nano 2.x) que se usa conectándola a una protoboard. La Protoboard es una placa de pruebas en los que se pueden insertar elementos electrónicos y cables con los que se arman circuitos sin necesidad de soldar ninguno de los componentes.

La placa Arduino Nano está inserta en el chasis del REP, como se indica a continuación en la Ilustración 36:

²⁰ Arduino: tecnología para todos. Disponible en: <http://arduino4dhtics.weebly.com/iquestqueacute-es.html>

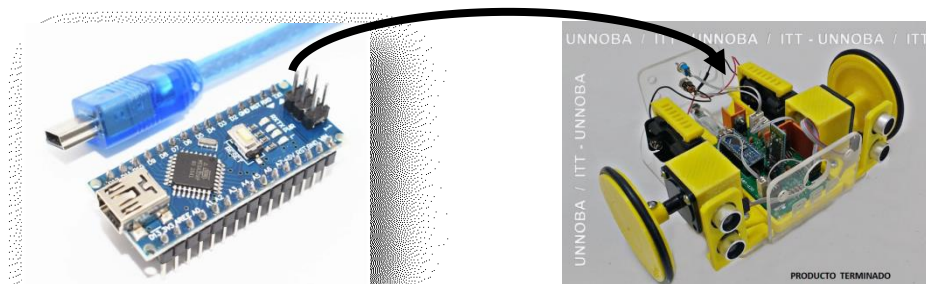


Ilustración 36

Sin embargo Arduino no se limita al hardware, sino que es una combinación de éste con software, ambos flexibles y fáciles de usar.

En la placa se encuentran conectados los controladores principales que gestionan los demás complementos y circuitos ensamblados en esta. Pero también requiere de un lenguaje de programación para su uso, por todo ello se la considera una herramienta completa.

En cuanto al hardware, la placa Arduino está diseñada con un micro controlador principal llamado Atmel AVR de 8 bits (que es programable con un lenguaje de alto nivel), presente en la mayoría de los modelos de Arduino.

Para comprender el funcionamiento será necesario saber qué sucede cuando en la interfaz el usuario escribe un programa, lo compila y termina materializándose en los movimientos del REP.

En primer lugar, el usuario de la interfaz crea un programa y ejecuta, para ello deberá hacer clic en el botón “Ejecutar/enviar” (Ilustración 37).

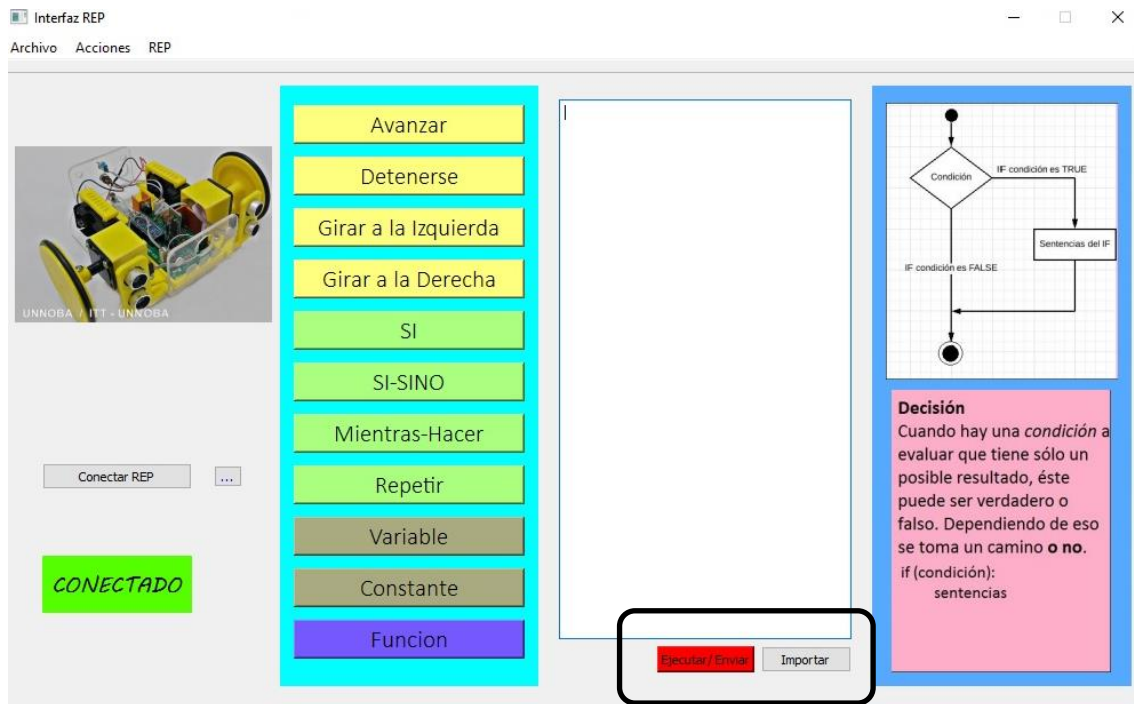


Ilustración 37: pantalla principal de la interfaz señalando el botón “Ejecutar/enviar”

En paralelo a lo anterior se confirma la conexión Bluetooth con el REP.

A continuación la interfaz “lee” para compilar el programa, y acto seguido se envía una cadena de ejecución en tiempo real al REP.

Finalmente para que el REP materialice los movimientos indicados por el programa del usuario, se utiliza el IDE (Integrated Development Environment) correspondiente. Se trata de un entorno de programación empaquetado, que incluye un editor de código, un compilador, un depurador y un constructor de interfaz gráfica; que para el caso de la placa Arduino es el software que puede descargarse desde el sitio web oficial de Arduino, ya que opera bajo licencia libre y está disponible a todo público. Su versión más reciente para todos los sistemas operativos es la versión Arduino 1.8.10. Una vez que el IDE de Arduino recibe la cadena de ejecución que le envió la interfaz, compila y ejecuta.

Ahora, si nos concentramos en la interfaz para el REP concretamente, al iniciar la aplicación se puede visualizar una ventana como la que se exhibe a continuación:

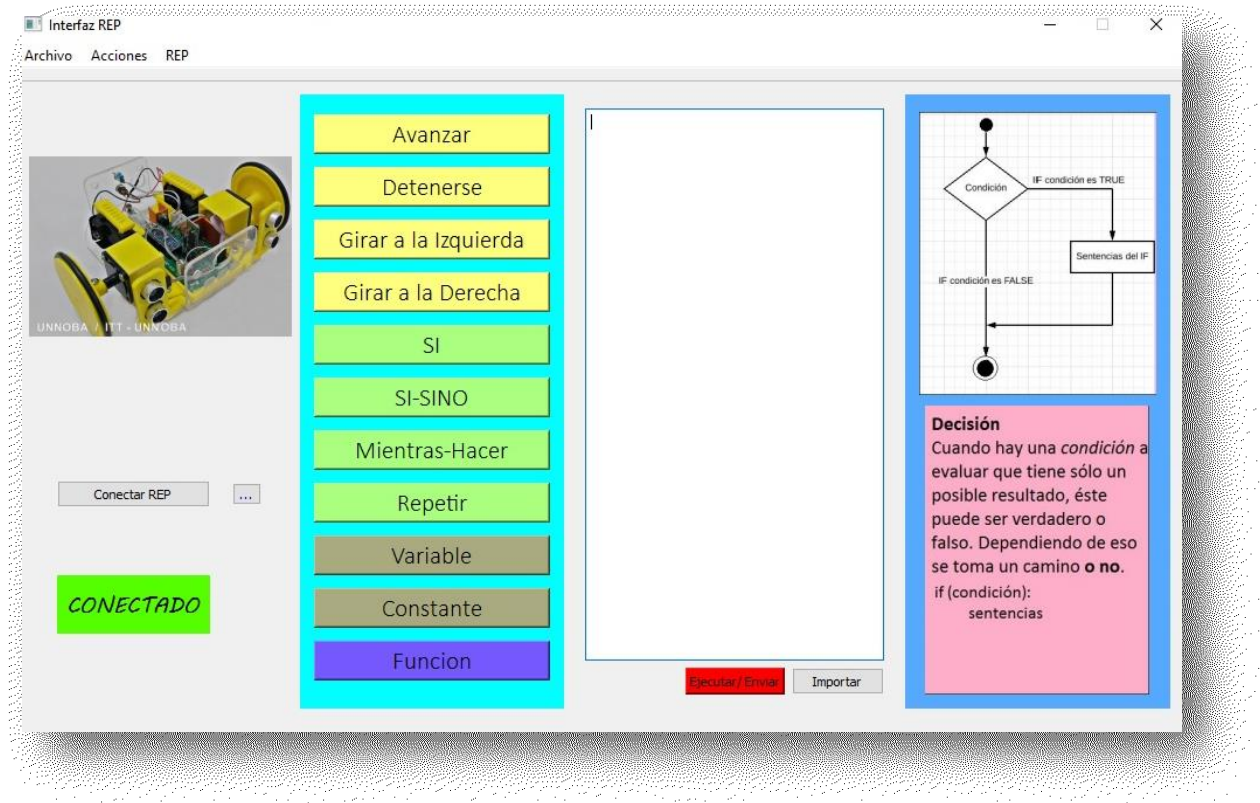


Ilustración 38: Captura de pantalla de la interfaz

En la Ilustración 38 puede observarse la interfaz diseñada con el objetivo de favorecer un trabajo introductorio en cuanto a conceptos y habilidades de programación. Se trata de una interfaz sencilla, desde lo visual, dado que se dividió a la ventana principal en seis paneles, tal como se visualiza en la Ilustración 39:

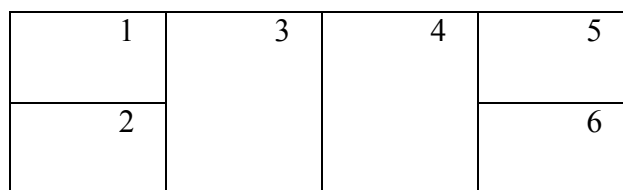


Ilustración 39: esqueleto de la interfaz, dividido en seis paneles.

En el primer panel (1) se encuentra una imagen del REP y en el segundo (2) se puede observar si el REP está o no conectado (observar la Ilustración 40).



Ilustración 40

Si el REP estuviera desconectado se deberá hacer clic en el botón “Conectar REP”. Con esta ayuda del sistema, el usuario puede verificar si el REP está listo o no, indicando de ese modo al usuario que podrá ejecutar el programa que desee diseñar.

En el tercer panel se muestra una serie de botones con instrucciones, estructuras de control, variables, constantes y funciones (observar la Ilustración 41).



Ilustración 41

La interfaz para REP trabaja con un pseudocódigo elemental que incluye las sentencias: Avanzar, Detenerse, Girar a la derecha, Girar a la Izquierda. Éstas son órdenes o instrucciones que el REP puede ejecutar.

A la hora de diseñar un programa estos pueden conformarse por una serie de sentencias que se ejecutan una tras otra, desde la primera hasta la última y siguiendo el mismo orden con el que aparecen. Estos programas tienen un flujo de ejecución estrictamente secuencial.

Por ejemplo si quisiéramos que el REP recorriera dos cuadras y luego girara a la derecha se podría escribir algo como lo que se muestra a continuación:

```
PROGRAMA
  INICIO
  Avanzar ()
  Avanzar ()
  GirarDerecha ()
  FIN
```

Gráficamente podría verse de la siguiente manera:

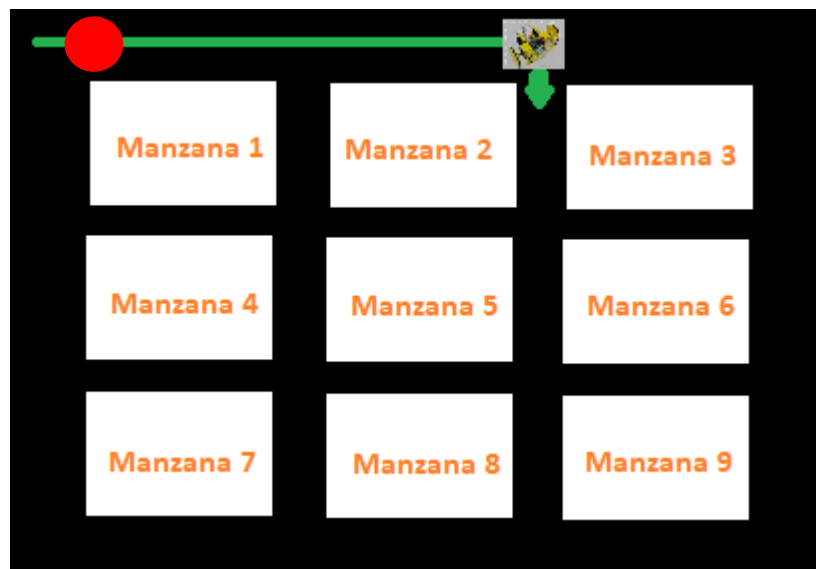


Ilustración 42: representación gráfica de un ejemplo.

De acuerdo al ejemplo anterior al REP se le está indicando que avance, luego que vuelva a avanzar y por último que gire hacia la derecha; por lo que ese será el único orden en el que se ejecutará el programa. Cabe mencionar que se debe interpretar a la Ilustración 42 como una pequeña ciudad en la que se moverá el REP. Particularmente esta ciudad está compuesta por nueve (9) manzanas, con los nombres: Manzana 1, Manzana 2, Manzana 3, Manzana 4, Manzana 5, Manzana 6, Manzana 7, Manzana 8 y

Manzana 9; y las líneas negras que pueden apreciarse en la ilustración serían las calles de la ciudad.

Gráficamente, si el REP estuviera ubicado en el lugar señalado con el círculo rojo (Ilustración 42) y luego se le indicara ejecutar el programa antes mencionado, el recorrido sería el trazado con la línea verde, finalizado girado hacia la derecha.

Por otra parte, para aquellos casos en los que fuera necesario alterar el flujo de ejecución para que pase de ser secuencial a permitir la toma de decisiones, y que en función de las mismas se ejecuten unas u otras sentencias, habrá que utilizar la estructura de control “Alternativa condicional”.

Por ejemplo, si se quisiera indicar al REP que recorra dos cuadras, luego que gire a la derecha y vuelva a recorrer una cuadra más, pero esta vez en el caso de encontrar algún obstáculo debiera detenerse; se podría escribir algo como lo que se muestra a continuación:

```
PROGRAMA
  INICIO
  Avanzar()
  Si ¿Hay obstáculo?
    Detenerse()
  Avanzar()
  Si ¿Hay obstáculo?
    Detenerse()
  Si no ¿Hay obstáculo?
    Girar a la derecha()
  Avanzar()
  FIN
```

En el caso puntual en el que se utilice la estructura de control SI (Alternativa condicional), se formulará una pregunta cuya respuesta será un valor de verdad, es decir la respuesta a la pregunta ¿Hay obstáculo? será: si o no, verdadero o falso; en otras palabras será un valor booleano.

Ahora bien, si el caso fuera solicitarle al REP que recorra dos cuadras por la calle 1 y luego una cuadra por la calle 6, si hubiera obstáculo en la esquina de las calles 1 y 6, el REP debería regresar al punto de partida. Caso contrario debe finalizar el recorrido. En donde el objetivo sería que el estudiante considere la utilización de la estructura de control “Alternativa condicional”, pero con el “SINO”. Puede esperarse algo como lo que se muestra a continuación:

```
PROGRAMA
  Avanzar()
  Avanzar()
  Si ¿Hay obstáculo?
    Detenerse()
    Girar a la derecha()
    Girar a la derecha()
    Avanzar()
    Avanzar()
  Sino
    Girar a la derecha()
    Avanzar()
```

Continuando con otras estructuras de control, si se solicitara al REP que resuelva el enunciado anterior, pero encontrando la manera de no escribir una y otra vez la misma línea de código (para el caso en el que alguna se repita); se podría escribir algo como lo que se muestra a continuación:

```
PROGRAMA
  Repetir 2-veces
    Avanzar()
  Si ¿Hay obstáculo?
    Detenerse()
    Repetir 2-veces
      Girar a la derecha()
    Repetir 2-veces
      Avanzar()
  Sino
    Girar a la derecha()
    Avanzar()
```

En la solución anterior se utiliza la estructura de control “Repetición simple”, que en la interfaz para el REP se visualiza con el botón: “Repetir”. En este caso vale utilizarla dado que la sentencia “Avanzar” se reitera más de una vez. Entonces para

evitar escribir una misma sentencia varias veces, y que esas repeticiones puedan ser ejecutadas una cantidad de veces predeterminadas, se utiliza esta estructura de control.

Por último, si se quisiera repetir una sentencia o un conjunto de estas, pero que esta vez esa repetición dependa de la evaluación de una condición, se utilizará la estructura de control “Repetición condicional”. Esta estructura, en la interfaz, se encuentra en el botón “Mientras-HACER” y, al igual que en la “Alternativa simple” depende de la evaluación de una condición que tendrá como resultado un valor booleano.

Suponiendo que se solicita al REP recorrer la Manzana 1 hasta que encuentre un obstáculo, para ese caso debería detenerse. Se podría escribir algo como lo que se muestra a continuación:

```
PROGRAMA
  INICIO
  Mientras no ¿Hay obstáculo? HACER
    Avanzar()
    Girar a la derecha()
  Detenerse()
FIN
```

Particularmente para este ejemplo habría que asegurarse que eventualmente el obstáculo va a aparecer, caso contrario nos encontraríamos en un bucle infinito. El bucle infinito en programación, ocurre cuando un ciclo se repite en forma indefinida suscitando un error lógico en el que el programa nunca termina. Salvado este error y retomando la estructura de control “Repetición condicional”, la interfaz del REP permite su uso para comprender el concepto.

Tal como se ha anticipado, los resultados de la ejecución del programa, se materializan en los movimientos y comportamientos que el REP realice en el piso del aula.

En el tercer panel de la interfaz se muestran los botones: variable, constante y función. En el contexto de una propuesta pedagógica, cuyo objetivo sea introducir a los estudiantes en la programación de computadoras, estos tres conceptos son fundamentales.

La “Variable” se define como el espacio en el sistema de almacenaje, que se identifica a través de un nombre simbólico y que puede almacenar ciertos datos, como así también modificarlos en el transcurso del programa. Particularmente en la interfaz para el REP las variables sólo pueden almacenar datos numéricos. Esta decisión responde a la pretensión de acotar el uso de la herramienta solo para introducir al estudiante en los conceptos de la programación. Sin embargo, puede ampliarse su uso a otros tipos de datos.

Considerando que se solicita al REP recorrer la Manzana 1, unas diez (10) veces pero utilizando la estructura de control “Repetición condicional”, podría escribirse algo como lo que se muestra a continuación:

```
PROGRAMA
  INICIO
  veces=0
  Mientras veces < 10 HACER
    Repetir 4 veces
      Avanzar()
      Girar a la derecha()
    veces= veces + 1
  Detenerse()
FIN
```

Si se observa el programa anterior, el valor que contiene la variable “veces” se incrementa en cada iteración, es decir “veces” comienza con un valor inicial cero (0) y va aumentando su valor en uno (1) en cada iteración del “Mientras- HACER”, hasta que finalmente cuando contiene el valor diez (10) REP se detiene.

Cuando se quiera trabajar el concepto de “Constante” como valor fijo y que, por lo tanto, no puede alterarse durante la ejecución del programa; se utilizará el botón con su homónimo en la interfaz para el REP.

Finalmente a la hora de abordar el concepto “Función”, en la interfaz, se hará “clic” en el botón indicado y se abrirá una nueva ventana en la que se puede programarla. La idea de “invitar” al usuario a escribir una función en una nueva ventana, responde a la intención de mostrar implícitamente que una función es una subrutina o subprograma que se agrupa con alguna lógica para resolver una tarea específica.

En las siguientes ilustraciones se puede apreciar la ventana que se abre cuando se escribe una función:

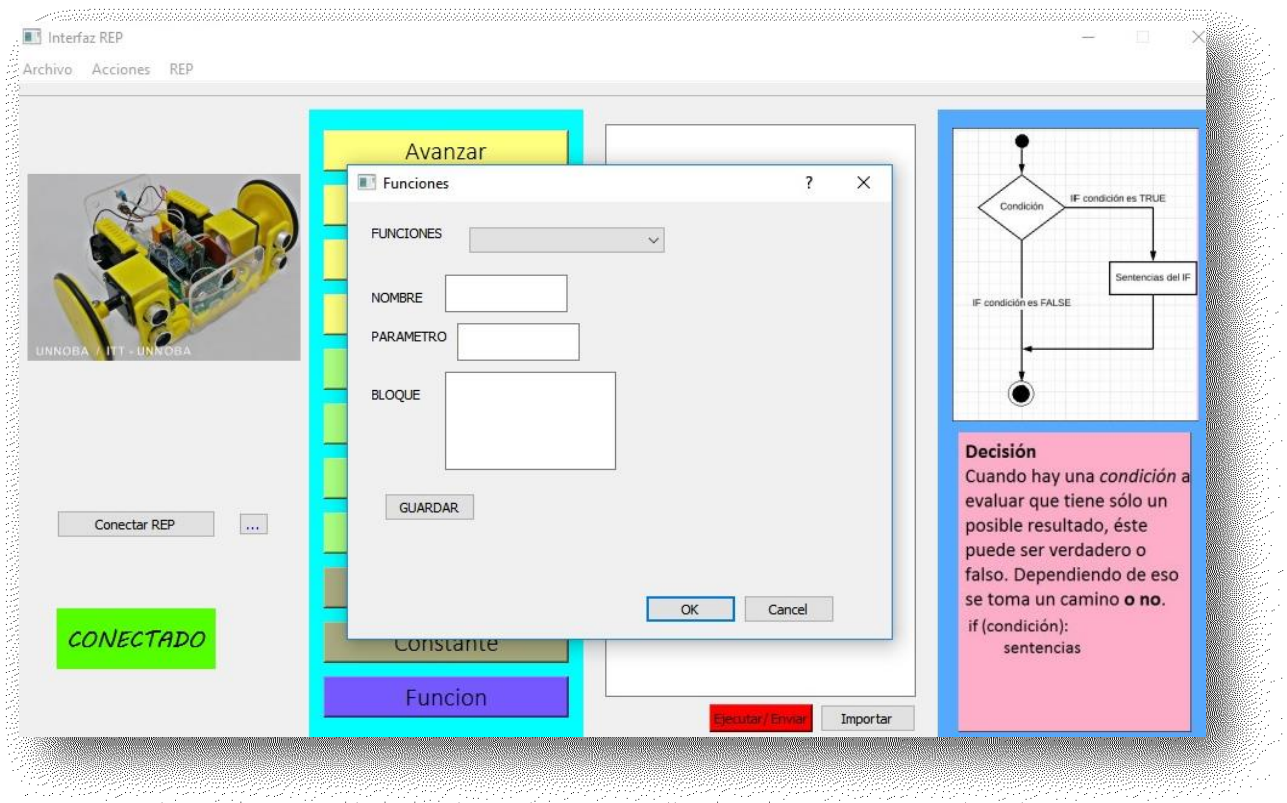


Ilustración 43: ventana de una función vacía.

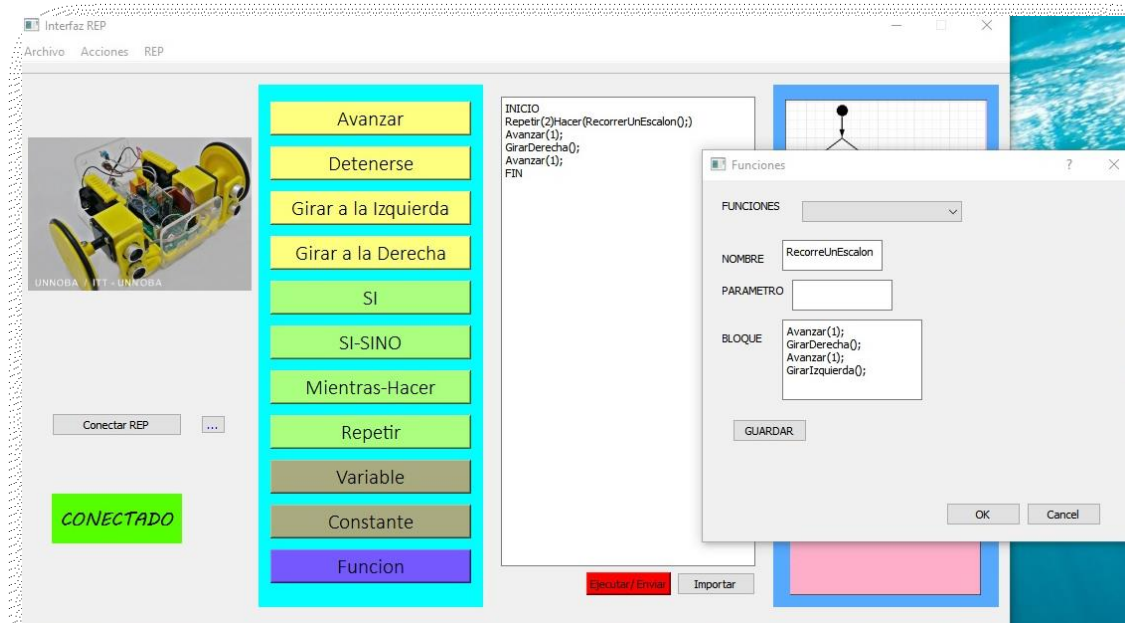


Ilustración 44: ventana de una función en estado de “construcción”.

Con todos estos elementos se pueden trabajar los conceptos básicos que deben desarrollarse en el inicio a la programación.

Continuando con la ventana principal de la interfaz, en el panel cuatro (4) se visualiza el programa que se va armando (observar la Ilustración 45).

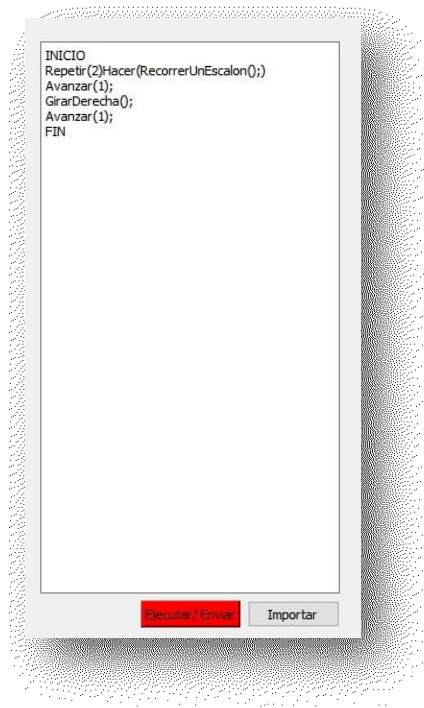


Ilustración 45

Los resultados de la ejecución del programa, se concretan en los movimientos y comportamientos que el REP despliegue en el piso del aula. La experiencia directa de visualizar la manera en que el REP responde a las órdenes que se le indican, transforma lo abstracto de las interfaces tradicionales de programación que simulan la ejecución de un algoritmo en pantalla, en un caso concreto.

En los paneles cinco y seis se podrá visualizar una ayuda de código en el lenguaje Python. Por ejemplo, cuando el usuario seleccione el botón “If”, en los paneles se mostrarán imágenes como las que se pueden apreciar en la Ilustración 46. Como se observa también, los textos de ayuda se presentan en el lenguaje Python. Sin embargo estas “ayudas” son sólo imágenes, pudiéndose incorporar otros lenguajes en función de las necesidades de los usuarios.

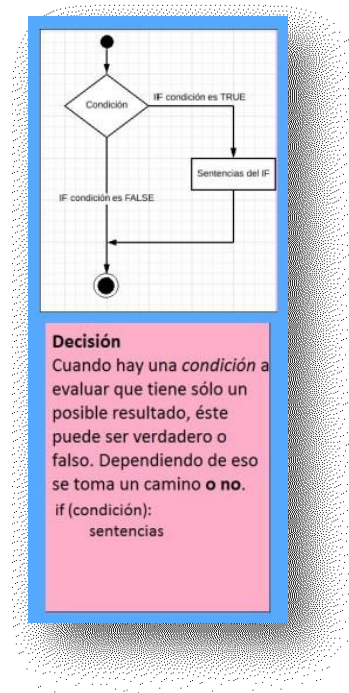


Ilustración 46

3.7 Justificación de la selección del lenguaje de programación

El robot está desarrollado con tecnología Arduino²¹, y la programación de este micro-controlador está basada en C++. Este es un punto a tener en cuenta para elegir el lenguaje de desarrollo de la interfaz, para poder hacer el traspaso de información entre ésta y el micro-controlador del REP de manera más fluida.

Otro ítem importante es el lenguaje que se utilizará en la interfaz, el cual en este caso es Python, ya que hay que tener en cuenta que esto también afecta a la complejidad del desarrollo a la hora de convertir las órdenes recibidas en Python en información para el REP.

Retomando la definición de interfaz, ese “adaptador” entre dos entidades que se comunican de forma distinta, para el caso de este trabajo de investigación será lo que transforme el código Python recibido en movimientos de nuestro REP.

²¹ Arduino: tecnología para todos. Disponible en: <http://arduinohtics.weebly.com/iquestqueacute-es.html>

Cuando se habla de código Python, se hace referencia al lenguaje de programación que utilizará el usuario del robot (en este caso el lenguaje que utilizarán los docentes para sus clases). En ese sentido el estudiante se relacionará con el robot, a través de Python, y esta elección se debe a que durante el año de ingreso a la UNNOBA se abordan los conceptos básicos de programación de computadoras con ese lenguaje.

En cuanto al uso educativo, se aspira a que los estudiantes estén más involucrados haciendo que sean constructores de su conocimiento de manera práctica y motivadora, al poder ver representadas sus teorías, investigaciones y creaciones en algo físico y concreto.

De esta manera, se pretende guiar al estudiante a desarrollar sus habilidades en diseño, programación, trabajo en equipo, construcción y testeo, entre otros aspectos, que se ven incluidos en el uso de REP.

Asimismo se busca aprovechar las habilidades y conocimientos que poseen los estudiantes de nivel superior que actualmente oscilan entre los 17 y 30 años de edad, lo cual refiere una edad que los convierte en Nativos Digitales.

CAPÍTULO 4. MARCO REFERENCIAL

El trabajo se desarrollará en el Instituto de Investigación y Transferencia de Tecnológica de la UNNOBA. El ITT cuenta con un espacio de 80 m² compuesto de catorce puestos de trabajo con notebooks, impresoras y sala de reuniones con proyector.

Además se tiene acceso a:

- Cuatro laboratorios equipados pertenecientes a Escuela de Tecnología de la UNNOBA.

- Cámara multiespectral Tetracam Lite.

- Sala CISCO equipada con:

- 4 Intel Pentium 4 2,40Ghz - 1 GB RAM - Disco 40 GB.

- 6 AMD Athlon II x2 245 - 2 GB RAM - Disco 160 GB.

- Racks: 6 Router Cisco1841 - 6 Switch Cisco Catalyst 2960.

- Grid compuesto por:

- 6 Intel Pentium 4 2,40Ghz - 1 GB RAM - Disco 40 GB.

- 1 Intel Pentium 4 2,40Ghz - 512 MB RAM - Disco 80 GB.

- Hp Compaq d220 MT - 512 MB RAM - Disco 40 GB.

- Dos Blade administrados por la Prosecretaria de TIC de UNNOBA:

- IBM eServerBladeCenter H - BladeServer H - 14 con redundancia al midplane y power - 2 fuentes 2980W PSU - 2 HS23 - 8 Xeon 4C E5-2609 2.40GHz/1066MHz/10MB - 48GB PC3-12800 CL11 ECC DDR3 1600MHz - 2x300GB 10K 6Gbps SAS 2.5" SFF HS SAS SED.

- IBM eServerBladeCenter H - BladeServer H - 14 con redundancia al midplane y power - 2 fuentes 2980W PSU - 2 HS23 - 2 Xeon 4C E5-2609 2.40GHz/1066MHz/10MB - 48GB PC3-12800 CL11 ECC DDR3 1600MHz - 2x300GB 10K 6Gbps SAS 2.5" SFF HS SAS SED.

- Cuatro salas de video conferencia con suministros para la interacción con otros equipos de investigaciones.

- Conectividad, Telefonía IP, file server, email provisto por UNNOBA.

Las pruebas en clase con los estudiantes se realizarán en el marco de la asignatura “Introducción a la Programación Imperativa”, que cuenta con un plantel docente conformado por siete (7) docentes en educación superior.

El equipo se conforma por dos (2) Analistas de Computación, dos (2) Programadoras Universitarias y tres (3) Licenciadas en Sistemas. Cabe aclarar que más adelante, en esta tesis, se pondrá en contexto respecto de la descripción de la asignatura, sin embargo por el momento se señalarán sólo algunos aspectos.

El plantel docente se divide en ambas sedes de la UNNOBA, trabajando una parte en la ciudad de Junín y otra parte en la ciudad de Pergamino.

Para IPI se abre una (1) comisión para la clase teórica en cada sede de la universidad, tres (3) tres comisiones de práctica en la sede de Junín y dos (2) para la sede de Pergamino. La diferencia en la cantidad de comisiones prácticas entre sedes se debe a que se cumple que siempre en Junín hay más inscriptos que en Pergamino.

Año	Inscritos	
2019	Total	216
	Junín	139
	Pergamino	77
2018	Total	216
	Junín	137
	Pergamino	79
2017	Total	169
	Junín	114
	Pergamino	55
2016	Total	141
	Junín	88
	Pergamino	53
2015	Total	138
	Junín	83
	Pergamino	55

Tabla 3: Cantidad de inscriptos de los últimos cinco (5) años en IPI.

CAPÍTULO N° 5. EXPERIMENTO

EL REP COMO ESTRATEGIA PARA PROMOVER HABILIDADES EN LA PROGRAMACIÓN IMPERATIVA

5.1 Estilos, estrategias y enfoques de aprendizaje

Partiendo de la idea que cada estudiante tiene una manera personal de acercarse al conocimiento, lo que determina modos distintos de aprender, se realiza inicialmente un diagnóstico orientado a interpretar las dificultades presentadas por los estudiantes en la asignatura IPI, como así también sus factores causales, para planificar posteriormente estrategias de enseñanza sobre la base del diagnóstico del procesamiento de información de los estudiantes.

Con tal finalidad se aplicó a los estudiantes una encuesta para determinar posibles explicaciones a estas dificultades (Ver Anexo I). De los 169 inscriptos en la materia, 60 estudiantes respondieron la encuesta en julio de 2017. En ese entonces, solo 34 de ellos habían aprobado IPI y se encontraban cursando PI (cifra que representa el 20,12 % de regularizados en relación a los inscriptos y el 36,56 % de regularizados en relación a los asistentes). La encuesta contaba con preguntas acerca de la dificultad de las tareas ejecutadas por los estudiantes, de la calidad del material pedagógico, de la incorporación y uso de recursos didácticos en clase, de sus estilos de aprendizaje y de sus hábitos de estudio. Todas las preguntas fueron respondidas en una escala de Likert de cinco valores.

En líneas generales los resultados de la encuesta indican que los estudiantes prefieren aprender involucrándose en actividades prácticas (dimensión sensitiva) y tienen preferencia por los materiales didácticos de actividades de aprendizaje relacionadas con la dimensión activa.

5.2 Ejecución del REP y su interfaz

5.2.1 Prueba piloto.

Durante el ciclo lectivo 2018, el cuatrimestre se planificó nuevamente en un espacio del EVEA de la universidad (UNNOBA Virtual) presentado con un formato configurado conforme a las intervenciones que las docentes querían realizar.

La configuración fue similar a la del año anterior, por ejemplo desde la primera clase se proporcionó el material referido al encuadre de la cátedra: “Programa de la asignatura” y “Documento de aprobación”, para contribuir a la planificación personal de cada estudiante. También en el curso se podía apreciar la conformación del equipo docente para ambas sedes, con sus respectivos horarios y fechas de comienzo de cursada. Seguidamente, en el apartado “Herramientas de trabajo” se compartió una presentación Prezi y dos infografías con las explicaciones necesarias acerca del uso de y la instalación de las herramientas a utilizarse durante la cursada.

Otra estrategia en el ámbito de la asignatura fue el contacto con docentes a través de un formulario de Google con el objetivo de mantener una comunicación constante y que los alumnos puedan realizar consultas o brindar sus opiniones con miras a mejorar la calidad del desarrollo del curso que, en líneas generales, fueron positivas (Ver Anexo II).

A continuación se pueden apreciar algunas preguntas con sus respectivas respuestas:

¿Comprendés los conceptos que explica la docente de teoría?

1 2 3 4 5

Se entiende muy poco Se entiende todo

Ilustración 47: pregunta dos

¿Comprendés los conceptos que explica la docente de teoría?

12 respuestas

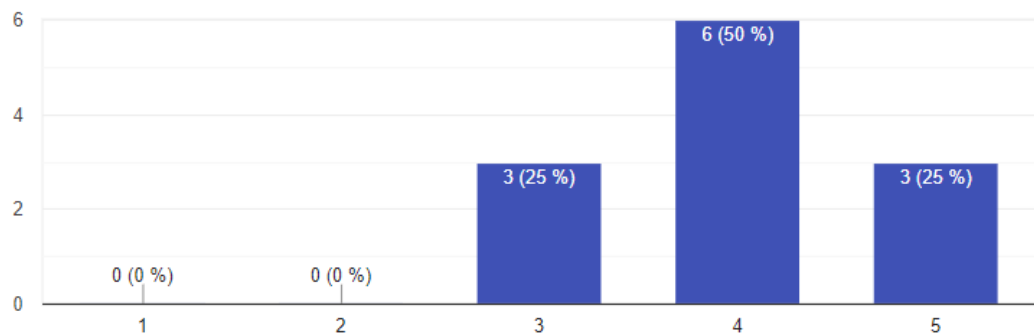


Ilustración 48: resumen de respuestas de la pregunta dos.

¿Comprendés los conceptos que explica la docente de práctica?

1 2 3 4 5

Se entiende muy poco Se entiende todo

Ilustración 49: pregunta tres

¿Comprendés los conceptos que explica la docente de práctica?

12 respuestas

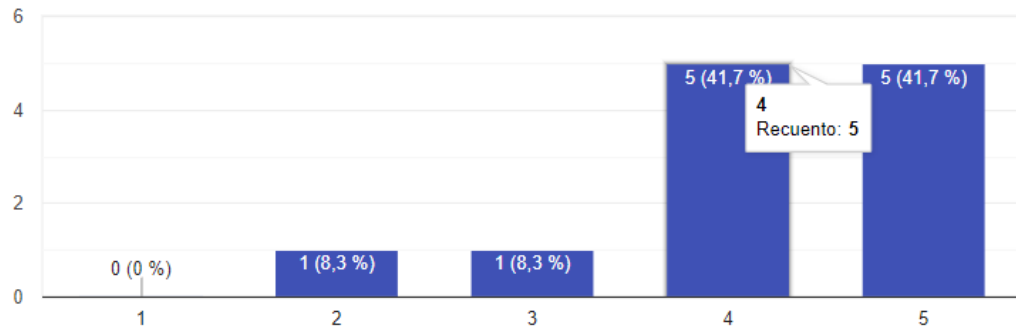


Ilustración 50: resumen de respuestas de la pregunta tres.

¿Qué actividades te resultan más útiles durante las clases? (marcá todas las que consideres)

- Explicaciones de las docentes de los temas
- Explicaciones de las docentes sobre ejercicios prácticos
- Tiempo que la docente nos deja para trabajar solos
- Consultar dudas de ejercicios que hice en mi casa

Ilustración 51: pregunta cuatro

¿Qué actividades te resultan más útiles durante las clases? (marca todas las que consideres)

12 respuestas

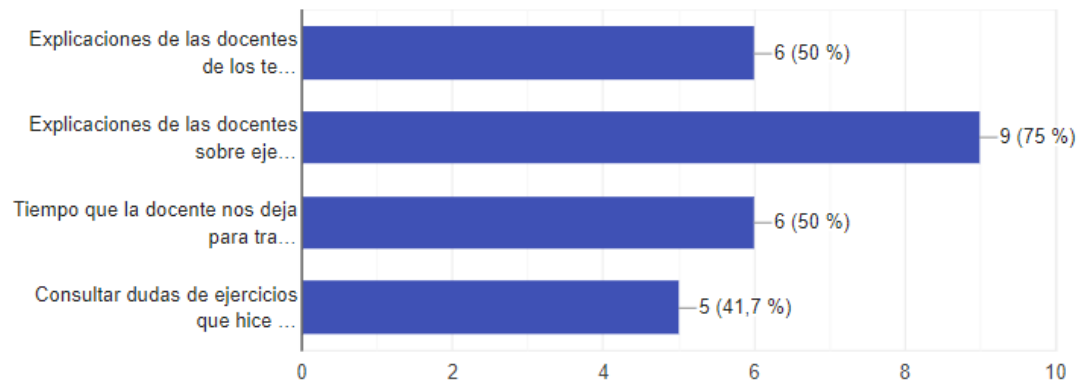


Ilustración 52: resumen de respuestas de la pregunta cuatro.

¿Considerás que podés llevar la asignatura al día?

- Sí
- Parcialmente
- No

Ilustración 53: Pregunta seis.

¿Considerás que podés llevar la asignatura al día?

12 respuestas

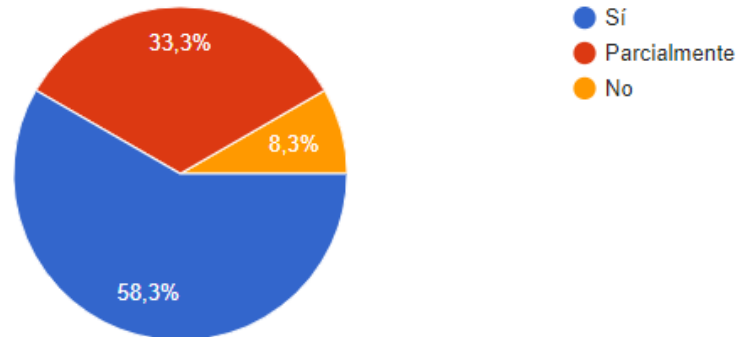


Ilustración 54: resumen de respuestas de la pregunta seis.

Con respecto al material teórico de la asignatura ¿te resulta accesible? *

- Sí
- Parcialmente accesible
- No

Con respecto a las prácticas de la asignatura ¿te resulta posible resolverlas con lo que ves en clase?

- Sí
- Parcialmente
- No

Ilustración 55: preguntas nueve y diez.

Con respecto al material teórico de la asignatura ¿te resulta accesible?

15 respuestas

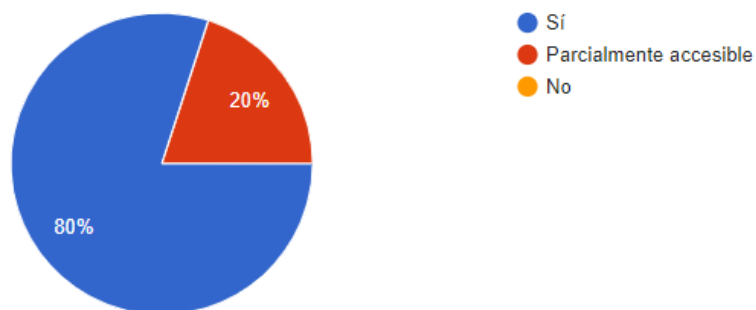


Ilustración 56: resumen de respuestas de la pregunta nueve.

Con respecto a las prácticas de la asignatura ¿te resulta posible resolverlas con lo que ves en clase?

11 respuestas

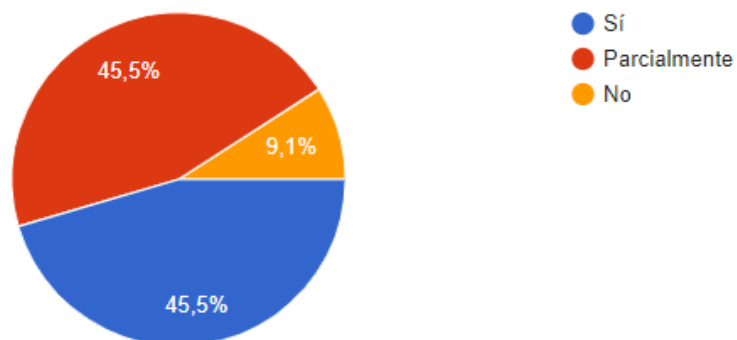


Ilustración 57: resumen de respuestas de la pregunta diez.

En dicho contexto, durante el primer mes de cursada de ese mismo año, se realizó una prueba en la que se propuso a los estudiantes de primer año de las carreras del área

de Informática una serie de ejercicios diseñados con la utilización del REP y su interfaz en el marco de la asignatura “Introducción a la Programación Imperativa”.

Los contenidos de la unidad temática en la que se realizó la experiencia incluyen: secuencia de instrucciones, funciones y repetición simple, alternativa condicional, repetición simple y repetición condicional.

Como ya se mencionó en el CAPÍTULO N°3. METODOLOGIA PROPUESTA punto 3.1 el Robot Educativo Programable (REP), este trabajo de tesis se enmarca bajo la línea Robótica e Interacción Hombre-Máquina (HCI), utilizando como herramienta al REP en la asignatura: “Introducción a la Programación Imperativa”. Con tal finalidad, se diseñó una interfaz para que el REP pudiera emplearse como herramienta para el caso de estudio seleccionado.

Esta experiencia involucró a un total de 138 estudiantes de la sede perteneciente a la ciudad de Junín. Durante ese mes se dictaron cuatro clases prácticas en las cuales las docentes a cargo de la cátedra plantearon diferentes ejercicios con un grado de dificultad creciente (ver Anexo III). El planteo de cada clase era tradicional, en el sentido de que la docente explicaba los conceptos teóricos para luego abordar los ejercicios de la práctica.

Para resolver los ejercicios, que pueden verse en el Anexo III, se podían utilizar las siguientes sentencias “comprendidas” por la interfaz del REP:

Avanzar()

Girar a la derecha()

Girar a la izquierda()

Detenerse()

En este sentido las anteriores eran las órdenes que la interfaz podría interpretar y de ese modo “ordenar” al REP ejecutar.

Además se consideraba el sensor: ¿Hay obstáculo? Es decir el “REP” podía preguntarse si efectivamente se encontraba frente a un obstáculo y de acuerdo a esa respuesta “decidir” qué hacer.

Como pre- condiciones se planteaban que: REP siempre se encuentra ubicado en la esquina de las calles 1 y 2 mirando por la calle 1, y siempre que hubiera obstáculos estarán en las esquinas.

Es importante señalar que, teniendo en cuenta las dimensiones del grupo clase se decidió dividir el trabajo en tres (3) comisiones. Se realizaron reuniones en un aula común de la universidad, donde en un primer encuentro se mostró la interfaz y el robot junto con sus instrucciones de uso a través de la aplicación, mencionando las sentencias que eran “comprendidas” y el modo de manejarlo a través de una PC.

Luego, en cada encuentro, se explicaron conceptos teóricos; se propusieron diferentes enunciados, a partir de los conceptos y habilidades planificados trabajar; y por último se los invitaba a pensar sus propias soluciones.

Al contar con un solo robot, una vez pensadas las soluciones, estas eran probadas con el REP. Seguidamente el grupo clase discutía la solución evaluando en conjunto cada propuesta, agregando, quitando o editando lo que fuera necesario.

La misma dinámica se repetía con algunas otras soluciones, dependiendo del tiempo con el que se contase, a los efectos de realizar una puesta en común y un análisis de las diferentes formas de abordar una solución a un mismo problema. La intervención docente consistía en guiarlos con comentarios o preguntas disparadoras.

Al finalizar estos encuentros se diseñó una evaluación en la que se pudieron observar resultados favorables. La misma consistía en una serie de enunciados,

similares a los que los estudiantes hubieran resuelto en los encuentros anteriores, es decir problemas que se resolvían a través del uso del REP.

Se obtuvieron los siguientes resultados:

- De 138 inscriptos en la asignatura, se presentaron a la evaluación 98 estudiantes, lo que representa un 71,01 % del total.

Históricamente en esta universidad los asistentes a la asignatura rondaban un promedio del 54,85 % de los inscriptos. Retomando los datos de las tablas 2 y 4, en el 2010 el porcentaje de asistentes fue del 60 %, en 2011 del 67 %, en 2012 del 52,76 %, 2013 del 52,76 %, 2014 del 54,41 %, 2015 del 54,35 %, 2016 del 42,55 % y en el 2017 del 55,03 %. Esto muestra un alto incremento de la asistencia, y por ello se infiere que el impacto de trabajar con el robot, tiene relación directa con ese incremento.

- De los presentes aprobó el 43,88%, lo que representa un 31,16 % en relación al total de inscriptos.

Nuevamente retomando los datos de las tablas 2 y 4, y analizando los porcentajes de estudiantes que regularizaron la asignatura sobre el total de asistentes, entre los años 2015 y 2017 se obtienen los siguientes resultados: 2015 del 33,33 %, 2016 del 48,33 % y en el 2017 del 36,56 %; lo que implica un promedio del 39,40 %.

Entonces si comparamos este número (39,40 %) contra el 43,88% de presentes que aprobaron la evaluación de la experiencia, vemos un considerable incremento; que puede leerse como un impacto positivo derivado de trabajar con el REP. Además si se calcula el promedio de estudiantes que regularizaron la asignatura sobre los inscriptos, también entre los años 2015 y 2017, se obtiene

un resultado del 19,60 %; que comparado con el 31,16 % de estudiantes que aprobaron la evaluación de la experiencia denota un alto incremento.

Acompañado a la evaluación se entregó a los alumnos un cuestionario autoadministrado donde debían responder algunas preguntas referidas al uso del REP y su interfaz. En líneas generales, se pudo determinar qué: el 80% de los estudiantes estuvo muy satisfecho con la intervención del REP y la interfaz, el 65% pudo aplicar los conocimientos obtenidos con el uso del REP en el examen, el 75% cree haber aplicado correctamente las técnicas utilizadas con el REP y el 50% cree haber aprobado el parcial.

Al respecto se puede destacar que tanto las evaluaciones como las encuestas arrojaron resultados favorables en relación a los años anteriores, y otros que luego de su respectivo análisis, permitieron la planificación de ciertos cambios en el uso de la interfaz y especialmente en las estrategias. Es decir, se replantea el uso del REP y de su interfaz a través de una didáctica diferente, la didáctica por indagación. Estos ajustes fueron incorporados a la experiencia, que tuvo lugar en el año 2019.

5.2.2 Experiencia 2019

Durante la cursada del 2019 en IPI se abordó como estrategia aplicar la didáctica por indagación en el comienzo de la cursada. Sin embargo, antes de comenzar con la experiencia, el equipo docente debió capacitarse en una nueva perspectiva y visión de la enseñanza de la programación, tal como se explicitó anteriormente en este trabajo de tesis.

En líneas generales, la propuesta pretende ubicar al estudiante en un rol más activo y al docente en uno más reactivo, para que aquél analice un problema (presentado por el docente) y plantee una o más posibles soluciones, apoyado en el bagaje de

conocimientos que trae consigo. Tal como exponen Sampson et al. a través del aprendizaje por indagación, los estudiantes deben encontrar soluciones a un problema a partir de un proceso de investigación, usualmente poniendo énfasis en el trabajo cooperativo y en la extracción de ideas a través de la reflexión sobre las actividades realizadas para construir la solución. De este modo, se invita al sujeto a recorrer su propio bagaje cognitivo para valerse de lo que "ya sabe" y en todo caso, ofrecerle nuevas herramientas para completar ese conocimiento.

La experiencia involucró a un total de 216 inscriptos, con un porcentaje de asistencia del 56,48 %, un porcentaje de regularizados en relación a los inscriptos del 22,22 %, y un 39.34 de regularizados en relación a los asistentes. Cabe destacar que en esta oportunidad la experiencia involucró estudiantes de ambas sedes, Junín y Pergamino, dadas la posibilidad de la tesista de trabajar en ambas ciudades.

En esta oportunidad no se realizó ninguna encuesta para indagar acerca del uso del REP, sino que se evaluaron los resultados finales de la asignatura, los cuales demostraron una mejoría considerable en términos generales.

5.2.3 Discusión de los resultados

Los resultados positivos arrojados por la encuesta realizada en el marco de la prueba piloto (2018) y la notable mejoría respecto a años anteriores, en lo que respecta a los porcentajes de aprobación de la evaluación dentro de la asignatura, permiten inferir que la utilización del REP y su interfaz han sido favorables. A su vez, las docentes del equipo observaron que la introducción al lenguaje Python fue menos compleja en términos de la articulación que se pudo realizar, dado que los estudiantes ya habían incursionado en una herramienta para el desarrollo de ejercicios en la programación

imperativa. Tanto los porcentajes de estudiantes regularizados sobre los inscriptos, como los regularizados sobre los asistentes se incrementaron (ver Tabla 4).

En ese sentido, se analizan los resultados finales de las cursadas de la asignatura en años anteriores, para compararlos con los resultados que brindaron las experiencias antes mencionadas.

Año	Inscriptas/os	Asistentes	Regularizadas/os sobre las/os inscriptos	Regularizadas/os sobre las/os asistentes
2019	216	56,48 %	22,22 %	39,34 %
2018	137	49,64 %	15,33 %	30,88 %
2017	169	55,03 %	20,12 %	36,56 %

Tabla 4: comparación de resultados finales de la asignatura, tomando para el año 2019 sólo estudiantes de la sede Junín

Año	Inscriptas/os	Asistentes	Regularizadas/os sobre las/os inscriptos	Regularizadas/os sobre las/os asistentes
2019	216	56,48 %	22,22 %	39,34 %
2018	216	47,22 %	15,28 %	32,35 %
2017	169	55,03 %	20,12 %	36,56 %

Tabla 5: comparación de resultados finales de la asignatura, tomando para cada año estudiantes de ambas sedes

La diferencia entre las tablas anteriores es que en la Tabla 4, fila correspondiente al año 2018 sólo se comparan los resultados finales de las cursadas de aquellos estudiantes con los que se vivió la experiencia antes mencionada (siendo éstos estudiantes cursantes en la sede de la ciudad de Junín); pero en la Tabla 5 se toma el total de los ingresantes, es decir estudiantes de ambas sedes.

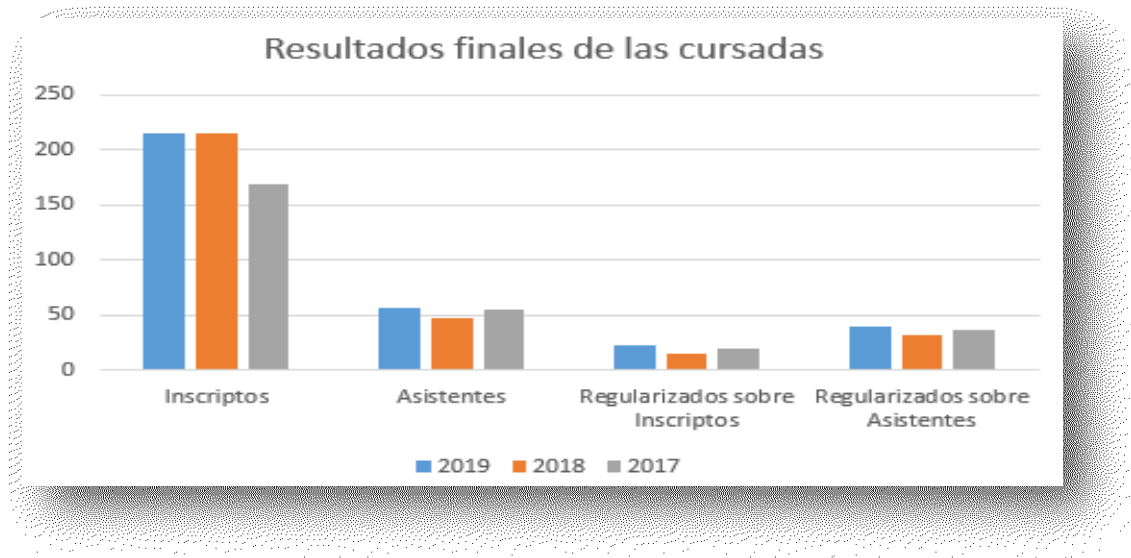


Ilustración 58

En la Ilustración 58, se observa que la asistencia en el año 2019 creció, como así también el número de estudiantes que logró regularizar la materia. Se infiere que tales resultados tendrían relación directa con el uso de tecnología en el aula, la cual efectivamente los motiva a estudiar y trabajar en lo que se propone en el marco de la asignatura. Todo esto sin olvidar que sin el sustento de una planificación que fundamente y acompañe su uso, no se asegura el éxito. Dicho en otros términos, la tecnología en el aula puede llegar a ser un gran motivador siempre y cuando esté inmersa en una propuesta educativa que le otorgue sentido.

Se estima que tales ensayos son muestras de las percepciones, que permiten reflexionar acerca de las experiencias en sí mismas, del uso que se le da a la interfaz y al REP, y de las posibilidades que pueden llegar a generarse a partir de su uso, pero principalmente al modo en que se los puede utilizar como herramientas educativas.

Es importante recordar que la propuesta se fundamentó en la didáctica de aprendizaje por indagación, la cual pretende situar en un rol más activo al estudiante y en uno más reactivo al docente. En ese contexto, este último se limita a presentar diferentes situaciones a resolver por el alumno sin introducirlo en ningún concepto

teórico. Luego, a medida que observa necesaria su intervención, aborda conceptos o trabaja destrezas que guíen al estudiante en la construcción de su conocimiento y la adquisición de habilidades.

Se puede resumir esta forma de trabajo del siguiente modo: se plantea un problema, se alienta a los estudiantes a pensar una estrategia de solución a resolver con la interfaz, se reflexiona sobre las soluciones planteadas (marcando aciertos, errores y las mejoras que pudieran existir), se brindan nuevas herramientas (de ser necesario), se prueban nuevas soluciones que pudieran llegar a surgir a partir de las intervenciones docentes, pudiendo finalmente volver a reflexionar sobre lo trabajado.

Es importante destacar que, si bien los estudiantes en un principio esperaban la clase tradicional, en términos de recibir toda una explicación del docente para luego realizar la transposición teórico práctica, recibieron satisfactoriamente la dinámica de trabajo y se apropiaron rápidamente de los contenidos.

5.2.4 Segunda experiencia 2019

Considerando que aún se pueden mejorar los resultados, se concretó en una tercera experiencia durante el mes de agosto y parte de septiembre del corriente año, con estudiantes que ya han cursado IPI pero que aún no han logrado regularizar la asignatura.

En este punto es necesario aclarar que, para incrementar la cantidad de estudiantes regularizados en IPI, se viene implementando desde hace unos años el dictado de lo que comúnmente se denomina “asignatura fuera de término”. Es decir, durante el segundo cuatrimestre se imparten clases a aquellos estudiantes que durante la cursada regular (primer cuatrimestre) no alcanzaron los objetivos para su aprobación.

Es por este motivo que se decide trabajar con este grupo en una nueva experiencia para aplicar nuevas estrategias que pudieran llevar a observar otros resultados. Los cambios radicaron en llevar adelante otro tipo de actividades como las dos que se describen brevemente a continuación:

1. En lugar de brindarles las consignas a los estudiantes serán ellos quienes diseñen nuevos enunciados en función de los conceptos y habilidades que se deseen trabajar. Un ejemplo podría ser: “diseñar un ejercicio en el que el programador deba utilizar una estructura de control repetitiva”, o “diseñar un ejercicio en el que el programador deba resolver un problema que pueda descomponerse en al menos dos funciones”. Luego se lo inducirá a que intente resolver el ejercicio para evaluar qué tan alcanzable es una solución tal y como él mismo lo planteó, para finalmente compartir su enunciado con un compañero quien intentará resolverlo.

2. Otra estrategia consiste en la “evaluación participativa por criterios”, cuyo objetivo es que los estudiantes puedan definir, conjuntamente con los docentes, las pautas que se tendrían en cuenta al momento de la evaluación de las actividades planteadas. La actividad involucra dos etapas, la primera radica en definir los criterios de evaluación y la segunda en utilizar dichos criterios para evaluar a los estudiantes.

Para resolver la primera etapa, se abrió un foro en el EVEA en donde todos los estudiantes pautaron los criterios de evaluación a la hora de rendir un examen en IPI. Para llevar adelante esta propuesta, se les dio pautas puntuales de cómo debatir en un foro para que el intercambio sea enriquecedor. Por ejemplo: a. en qué foro y con qué restricciones temporales se lleva a cabo el debate. b. todos los participantes deben realizar aportes y opinar sobre los de sus compañeros.

Las docentes, tras analizar los aportes, realizaron una devolución en un plazo predeterminado. Al culminar dicho plazo, se publicó una lista con todos los criterios definidos para el grupo clase.

En cuanto a la segunda etapa, una vez que quedaron definidos dichos criterios, los docentes hicieron entrega de un ejercicio a resolver con el REP y su interfaz. Este ejercicio fue resuelto por cada estudiante de manera individual, durante el encuentro presencial y evaluado de acuerdo a los criterios por ellos especificados. Tales criterios estaban disponibles y a la vista de todos, generando en los estudiantes una sensación de seguridad al estar conscientes bajo que pautas serían evaluados.

5.2.5 Discusión de los resultados

Se realizó un registro de la actitud y motivación del estudiante antes de iniciar la experiencia y una vez concluida la misma. El objetivo fue medir el resultado de la experiencia en relación con la metodología de evaluación desarrollada oportunamente.

Aspectos a evaluar:

- Grado de satisfacción con la experiencia de evaluación participativa por criterios.
- Percepción del nivel de dificultad con que se definen los criterios de evaluación.
- Aporte de la experiencia al proceso de enseñanza.
- Nivel de motivación del estudiante.

De acuerdo con el análisis de los resultados, en la medición realizada antes de la experiencia, se pudo establecer que el 85% de los estudiantes estaban de acuerdo con cambiar la metodología de evaluación, manifestando que la experiencia resultaría más interesante que la evaluación tradicional. Destacaron además su interés en la definición conjunta de criterios de evaluación entre pares y docentes, presuponiendo esto como una ventaja para la evaluación final. El 11% de los estudiantes estuvo en desacuerdo con estas apreciaciones y el 4% restante no contestó.

En este sentido, se pudo apreciar que la predisposición de la mayoría ante el grado de satisfacción que podría generar la experiencia de evaluación participativa por criterios, fue positiva.

Al analizar los resultados, luego de implementar la metodología de evaluación planteada en este trabajo, se pudo establecer que si bien la experiencia fue enriquecedora, ya que se logró un intercambio y cooperación entre los participantes, la mayoría de los estudiantes que presuponían una ventaja el hecho de definir los criterios

de evaluación, reconocieron que habían puesto pautas muy estrictas, lo que no los favoreció en el resultado de la evaluación.

CAPÍTULO N°6. DISCUSIÓN FINAL DE LOS RESULTADOS

Es importante comenzar observando que estos resultados se obtuvieron en el marco de diferentes experiencias desarrolladas a través del uso del REP y su interface, por lo que se puede inferir que el uso de estas herramientas ha favorecido el éxito de cada experiencia.

Los participantes ya tenían una evaluación positiva de la “asignatura fuera de término” de acuerdo a la encuesta inicial. En la encuesta sobre el perfil de los estudiantes se identificó una marcada preferencia por las actividades de aprendizaje de tipo prácticas, lo que permite predecir este resultado. También, los alumnos mostraron estar más satisfechos después de haber finalizado las clases, aumentando de esta forma el porcentaje de adhesiones con la metodología propuesta.

De igual modo, durante la primera experiencia realizada en el año 2019, se puede inferir que la metodología de aprendizaje por indagación y la aplicación de estrategias de enseñanza colaborativa, en materias vinculadas a la programación, promueven en los alumnos, un compromiso más fuerte y activo. El uso de los ambientes colaborativos les ofrecen la posibilidad de beneficiarse del conocimiento y habilidades del resto de integrantes del grupo, mejorando así sus propias destrezas (Jurado *et al.*, 2012) y el autoaprendizaje.

El trabajo colaborativo está asociado al conjunto de métodos de instrucción y entrenamiento, que buscan promover el desarrollo de habilidades como aprendizaje, desarrollo personal y social; y la responsabilidad no solo a nivel individual sino colectiva (Lucero, 2003). Por otra parte, la colaboración en el ámbito académico, nace y

responde a un nuevo contexto (social y cultural), donde se define el “cómo aprendemos” (socialmente) y “dónde aprendemos” (en red). (Zañartu, 2003).

Teague (2008), sostiene que la aplicación de estrategias de enseñanza colaborativa, en materias vinculadas a la programación, promueven en los alumnos, un compromiso más fuerte y activo, los alienta a pensar en voz alta y a verbalizar el proceso de resolución de problemas, satisfaciendo de esta forma, la necesidad de interacción de los alumnos actuales.

PARTE V. INFORME FINAL

A continuación se presenta un documento que explica en qué medida fueron alcanzados los objetivos propuestos, la validación o refutación de hipótesis, las estrategias áulicas en lo referido a la Programación Imperativa, las sugerencias y recomendaciones, y las futuras líneas de investigación.

Tomar consciencia de los vertiginosos cambios del contexto social, particularmente del tecnológico, requiere indefectiblemente advertir la necesidad de una transformación permanente del sistema educativo, y por consiguiente de la tarea docente en su conjunto. En este sentido los estudiantes reclaman, de forma explícita o implícita, nuevas estrategias didácticas que sean acordes a los tiempos que corren y a sus necesidades académicas y profesionales.

Por tal motivo, la primera instancia del presente trabajo de tesis, se encamina a revisar y evaluar las diferentes estrategias implementadas para la enseñanza de la programación imperativa, en la Universidad Nacional del Noroeste de la Provincia de Buenos Aires. En este sentido se ha incursionado sobre el modo en el que históricamente se viene trabajando en la asignatura Introducción a la Programación Imperativa, puntualmente en lo referido a: herramientas utilizadas, lenguajes de programación trabajados, bibliografía propuesta, diseño de los materiales didácticos, evaluaciones administradas, y demás aspectos que permitieron realizar un estudio

pormenorizado del encuadre metodológico de esta asignatura dictada en los primeros años de las carreras del área de informática.

La revisión y evaluación de la metodología de trabajo llevada a cabo hasta el año 2017; fue de vital importancia para poder finalmente repensar y reorientar dichas estrategias en pos de alcanzar nuevos y mejores resultados. Dicho en otros términos, el estudio de estos antecedentes justificó la formulación e introducción de cambios y de adaptaciones en pos de alcanzar mejores resultados en términos de cantidad de regularizados en la materia y, por consiguiente, de una mejor calidad de la propuesta educativa.

En respuesta a lo anterior y pensando en nuevas estrategias no abordadas hasta el momento en la mencionada Universidad, se decide incursionar sobre el uso de robots como herramienta educativa en el nivel superior. Es así entonces que se llevan a cabo diferentes experiencias en el marco de la asignatura IPI, las cuales permitieron alcanzar resultados prometedores y, desde luego, mejorables. Para estas experiencias se ha diseñado una interfaz para el uso del Robot Educativo Programable, actividad que se ha explicitado en el apartado N°3 de la presente tesis.

Por consiguiente se ha determinado que, para el estudio de caso de la materia “Introducción a la Programación Imperativa”, el uso del REP a través de la interfaz diseñada para este trabajo ha sido un aporte valioso por varios factores. Por un lado, el equipo docente que trabaja en la asignatura ha manifestado que resulta más atractiva la articulación entre el uso del REP y su interfaz y Python como lenguaje de programación utilizado en la asignatura. A su vez, las evaluaciones administradas y del trabajo áulico revelan que los estudiantes han podido apropiarse de habilidades para realizar procesos mentales de abstracción a través del diseño y ejecución de programas que ejecuta el REP.

Es importante reiterar que en el marco del presente trabajo de tesis, no se concibe a la tecnología o, mejor dicho a su uso, como la panacea del éxito en una propuesta educativa. Por el contrario se asume que tal éxito se encuentra ligado a un conjunto de elementos que, combinados estratégicamente, devienen en una conquista educativa. Es decir, no se concibe al uso de la interfaz y del REP como la solución a la problemática de la baja tasa de regularizados en la asignaturas, si no se acompaña de una propuesta educativa planificada en un contexto que estratégicamente brinde el uso de la tecnología como una herramienta más para el aprendizaje de la programación imperativa.

Finalmente se considera que con este trabajo de tesis perteneciente a la “Maestría en Tecnología Informática aplicada a la Educación” de la Facultad de Informática de la Universidad Nacional de La Plata, se ha logrado establecer nuevas estrategias educativas con respecto al paradigma imperativo de la programación de computadoras utilizando como herramienta al Robot Educativo Programable (REP).

Entre las principales estrategias se destacan las siguientes:

1. **Incorporar al REP y su interfaz** como instrumentos tecnológicos no tradicionales a las actuales propuestas educativas de dicha universidad.

Teniendo en cuenta los resultados respecto a los porcentajes de regularizados de los últimos años, si se comparan los porcentajes de asistencia, se puede observar que en el año 2017 hubo un número mayor de asistentes que en el 2018, lo que puede devenir en menores posibilidades de intervención docente. Si bien no se estudió la razón por la que disminuyó tan notablemente este número de asistentes a la cursada, si se intentó cambiar algunas cuestiones que pudieran hacer más atractiva la propuesta en IPI. Por ello en el año 2018 y nuevamente en el 2019 se llevan a cabo trabajos mediados por REP y su interfaz. En este último año la asistencia creció notablemente, por lo cual se

infiere que la aplicación de esta estrategia pudo ser parte del “atractivo” de la clase y por ello se hace una lectura positiva de la experiencia.

A su vez los porcentajes de aprobación, ya sea de los asistentes a clase como de los inscriptos, crecieron considerablemente en el año 2019.

2. Utilizar la **didáctica por indagación** como estrategia áulica.

De acuerdo a lo mencionado en el punto anterior, y en base a todo lo investigado al respecto, se recomienda el uso de la didáctica por indagación como estrategia áulica válida para la asignatura IPI en la UNNOBA. Esta didáctica fomenta la creatividad en términos de colocar al estudiante en un lugar en el que deba elaborar una solución a un problema sin previa intervención docente. También al momento de socializar sus resultados y cotejarlos con los de sus pares, se apela al pensamiento crítico y la reconstrucción de la información.

Por otra parte, esta estrategia coloca al estudiante en un rol de autoevaluador de sus propias producciones como así también de la de sus pares. Esto implica tener que verificar y valorar lo propio y lo ajeno, nutriéndose de esa situación a través de corrección de errores, valoración del esfuerzo, entre otras cuestiones.

3. Posicionar al **docente en un rol** más pasivo, de acompañamiento. Íntimamente vinculado a lo anterior, se sugiere que el docente que “se coloque a un lado” del proceso de aprendizaje. Esto implicaría que haga las veces de estrategia en cuanto a proporcionar apoyo para la realización de actividades, el abordaje del material didáctico, el uso de las diferentes herramientas; pero todo ello desde un lugar más reactivo y en función de las necesidades individuales de los estudiantes y las demandas generales del grupo.

Se sugiere que puntualmente que en las clases teóricas, donde históricamente la explicación se encontraba autocentrada en la figura del docente, y el abordaje de las

unidades temáticas se hacía desde una vertiente conceptual; se propicie un espacio en el que sean los estudiantes quienes descubran y construyan el conocimiento. Lo anterior no significa la ausencia de la figura docente sino que por el contrario, lo invita a reconfigurar su intervención bajo una estrategia que se adapte al ritmo y estilos de aprendizaje de sus estudiantes.

4. Posicionar al **estudiante en un rol** de mayor protagonismo. De cara a lo anterior, en donde se sugiere un docente en un rol más pasivo que conduce al estudiante a ocupar un rol más relevante en términos de autoaprendizaje, es fundamental planificar actividades que sitúen al alumno en un lugar de mayor protagonismo, en cuanto a su rol como aprendiz y también en un lugar en donde se valore su bagaje de conocimientos.

Se considera que la aplicación de la didáctica por indagación puede llegar a propiciar las condiciones necesarias para que pueda acercarse con facilidad a este rol protagónico. En donde sea él mismo quien analice diferentes situaciones problemáticas y plantee una o más posibles soluciones, apoyado en el bagaje de conocimientos que trae consigo.

5. **Comenzar trabajando con un pseudocódigo**, en lugar de hacerlo con un lenguaje de programación. Se entiende que para lograr introducir los conceptos básicos de la programación imperativa, resolver problemas mediante un algoritmo, modularizar y parametrizar y aplicar el paradigma estructurado en la resolución de problemas y correctas estructuras de control; puede presentarse el uso de un nuevo lenguaje (como lo es un lenguaje de programación) como una dificultad más en ese proceso. De modo que el uso de un pseudocódigo sencillo como los que se han trabajado en IPI, recordemos “El mundo de Ada” en el 2017 y la interfaz de REP en el 2018 y 2019, ha permitido una articulación más apacible entre ese pseudocódigo y el lenguaje de programación utilizado en la asignatura. En ese sentido se cree que es una estrategia a tener en cuenta.

Para concluir, y corroborando las hipótesis del presente trabajo, se obtiene que el uso de robots es de gran potencial para facilitar la aplicación más tangible y concreta de las habilidades de programación. Asimismo, la interfaz para el uso del REP permite establecer una forma diferente de acercar a los estudiantes a los contenidos, y motivarlos mediante el uso de un robot, siempre y cuando, a través de la intervención del docente, se logre un diseño de actividades motivadoras y estimulantes.

En cuanto al uso educativo, se obtiene que los estudiantes se muestran más involucrados al ser constructores de su conocimiento de manera práctica y motivadora, al poder ver representadas sus teorías, investigaciones y creaciones en algo concreto y tangible.

RECOMENDACIONES.

De la segunda experiencia del trabajo que se realizó con REP y su interfaz en la recursada de IPI aún no se pueden observar resultados dado que el cuatrimestre finaliza en el mes de diciembre. Sin embargo se concretaron varias dinámicas de trabajo novedosas en relación al modo en el que se venía trabajando en IPI, una de ellas fue la actividad en la cual los estudiantes debían diseñar nuevos enunciados en función de los conceptos y habilidades que se desean trabajar. Este ejercicio debía ser resuelto por el mismo alumno a los fines de evaluar qué tan alcanzable es la solución y como él la planteó, para finalmente compartir su enunciado con un compañero quien intentará también resolverlo.

Este tipo de dinámica implica incorporar estrategias que promuevan la construcción del saber por parte del individuo y en interacción con su entorno (ya que el conocimiento se construye durante las interacciones sociales). Esta estrategia es escogida debido a su potencial para ayudarlos a desarrollar habilidades de trabajo en

equipo y el auto aprendizaje que son de gran valor en un contexto profesional. Así, intercambiando sus roles, los estudiantes pueden ser capaces de aprender de las acciones y estrategias de sus compañeros.

También se ha mencionado la actividad de “evaluación participativa por criterios”, que invita al estudiante a adelantarse al proceso evaluativo definiendo aquellos criterios con los que será evaluado.

Es indispensable continuar trabajando en pos de poder formar competencias que, además de responder a los objetivos académicos de la asignatura, sean de vital importancia para la formación integral de un futuro profesional de las ciencias de la computación. En relación a esto es preciso trabajar la empatía, la comunicación, la confianza, el compromiso, la proactividad y la creatividad. Todas estas competencias son vitales no sólo durante su trayectoria académica sino también durante su desempeño profesional, y es por ello que desde el comienzo la Universidad debe enfocarse en potenciarlas.

FUTURAS LÍNEAS DE INVESTIGACIÓN.

A la luz de los favorables resultados obtenidos en cuanto a la incorporación el Robot Educativo Programable en las actuales propuestas educativas de la programación imperativa como herramienta pedagógica, se insta a continuar probando este instrumento, registrando sus resultados y evaluándolos con el fin de mejorar el dictado de la materia. De este modo se podrá mejorar la calidad educativa en la UNNOBA en el área en donde se trabaja con competencias relacionadas con el pensamiento computacional.

Por otra parte, se espera avanzar en el diseño de la interfaz incorporando nuevas funcionalidades entre las cuales se aspira poder desarrollar una versión para teléfonos smartphones; así como la posibilidad de conectar la interfaz con más de un REP.

Continuar la investigación de los sensores de REP en busca de poder diseñar una herramienta accesible, por ejemplo trabajar con sonidos para personas ciegas y luces para personas sordas.

Otro aspecto fundamental a trabajarse a los fines de compartir las experiencias de la UNNOBA es la difusión de los resultados de esta investigación y de los que se registrarán de acá en más.

REFERENCIAS BIBLIOGRÁFICAS

Álvarez Ruiz, A., Ron Zorzano, R., Núñez, P. (2013). Los efectos del marketing digital en niños y jóvenes: Smartphones y tablets ¿enseñan o distraen? Madrid: ESIC.

Areepattamannil S. (2012) Effects of inquiry-based science instruction on science achievement and interest in science: Evidence from Qatar. *The Journal of Educational Research* 105(2), 134–146.

Banchoff Tzancoff , C. Martín, S., Gómez, S. y López, F. (s.f). Experiencias en robótica educativa - Diez años trabajando con escuelas argentinas. LINTI: UNLP La Plata, Argentina. Recuperado el 07 de septiembre de 2020 de: http://sedici.unlp.edu.ar/bitstream/handle/10915/90729/Documento_completo.pdf-PDFA.pdf?sequence=1&isAllowed=y

Bartolomé, P. (2018, 17 de enero). La meta-cognición: ¿Por qué es útil conocer cómo funciona tu mente? *CogniFit. Salud, Cerebro y Neurociencia*. Recuperado el 10 de mayo de 2019 de: <https://blog.cognifit.com/es/que-es-metacognicion-definicion-funciones-ejercicios-consejos/>

Bevins, S., Price, G. (2016) Reconceptualising inquiry in science education. *International Journal of Science Education* 38(1), 17-29.

Clements, D. H. (1986). Effects of Logo and CAI environments on cognition and creativity. *Journal of Educational Psychology*, 78(4), 309-318.

Couso D., Jiménez M. P., López-Ruiz J., Mans C., Rodríguez C., Rodríguez J.M., Sanmartí, N. (2011) Informe ENCIENDE: Enseñanza de las Ciencias en la Didáctica escolar para edades tempranas en España. Madrid: Rubes Editorial. Recuperado el 13 de

septiembre de 2020 de: https://www.cosce.org/wp-content/uploads/2011/03/Informe_ENCIENDE.pdf

Costaguta, R. y Gola, M. (s.f). Identificación de Estilos de Aprendizaje Dominantes en Estudiantes de Informática. Recuperado el 2 de mayo de 2019 de <https://pdfs.semanticscholar.org/5575/be1010c61427cbaf03a7f3669fc1865019d9.pdf>

De Vicente, M. (03/10/2017). Aprendizaje Cognitivo: Tipos de aprendizaje, una guía educativa. *CogniFit. Salud, Cerebro y neurociencia*. Recuperado el 10 de mayo de 2019 de: <https://blog.cognifit.com/es/aprendizaje-cognitivo-tipos-aprendizaje/>

Figueroa, N., Cataldi, Z., Méndez, P., Rendón Zander, J., Costa, G. y Lage, F. (s.f). Los estilos de aprendizaje y las inteligencias múltiples en cursos iniciales de Programación. Recuperado el 2 de abril de 2019 de http://sedici.unlp.edu.ar/bitstream/handle/10915/21115/Documento_completo.pdf?sequence=1

Freiberg Hoffmann, A., Berenguer, D., Fernández Liporace, M., Ledesma, R. (2017). Estilos, Estrategias y Enfoques de Aprendizaje en Estudiantes Universitarios de Buenos Aires. *Psicodebate*, 17 (1), 9-34.

Garcíz Aretio, L., Ruiz Corbella, M., García Blanco, M. (2009). Claves para la educación: actores, agentes y escenarios en la sociedad actual. Madrid: Narcea SA y Universidad Nacional de Educación a Distancia.

Gutiérrez, E. Yuste, T y Fustes, M. (2011). Buenas prácticas en el desarrollo de trabajo colaborativo en materias TIC aplicadas a la educación. *Profesorado* 15 (1), 180-194.

Hernández Sampieri, R. Fernández Collado, C. Baptita, L (2014). *Metodología de la investigación* (6° ed.). México D, F., México: MacGraw-Hill education

Juárez Landín, C., Sánchez Ramírez, J.L., Mendoza Pérez, M.A., Sánchez Soto, J.M., (2016). Robots Bioloid y Nao como estrategia pedagógica para formar competencias en los estudiantes de Licenciatura. *Pistas Educativas*, 38 (122), 259-274.

López Ramírez, P.A. (2013). Aprendizaje de y con robótica, algunas experiencias. *Revista Educación* 37(1), 43-63.

McConney A., Oliver M.C., Woods-McConney A., Schibeci R., Maor D. (2014) Inquiry, Engagement, and Literacy in Science: A Retrospective, Cross-National Analysis Using PISA 2006. *Science Education* 98(6), 963-980.

Maestri, G. (2002). Creación digital de personajes animados: técnicas avanzadas. Madrid: Anaya Multimedia.

Muñoz, R., Noël, R., Barría, M., Barcelos, T., Silveira, I. 2015. Diseño y evaluación de un taller de robótica basado en Estilos de Aprendizaje para la enseñanza de Fundamentos de Programación. Recuperado el 25 de marzo de 2019 de <http://dx.doi.org/10.18682/pd.v17i1.626>

Odorico, A.H., Lage, F., y Cataldi, Z. (s.f). La robótica: Aspecto clave de la producción moderna vista desde una perspectiva pedagógica. Recuperado el 18 de abril de 2019 de http://sedici.unlp.edu.ar/bitstream/handle/10915/20833/585-La_rob_tica_WICC.pdf;jsessionid=E40754DB09650A7ED324FA95B25C1DDF?sequence=1

Pérez Angulo, J., y Pedroza Palomar, O. (2018). LM1: una metodología de estudio para la asignatura “Programación 1”. *Educere*, 22 (73), 635-648

Pina, A. (2014, 19 de Agosto). La robótica educativa ayuda a los alumnos a razonar; eso vale para Informática y para Filosofía. *Eldiario.es*. Recuperado el 29 de

noviembre de 2016 de https://www.eldiario.es/norte/navarra/ultima_hora/robotica-educativa-alumnos-Informatica-Filosofia_0_293621134.html

Pinto Salamanca, M.L., Barrera Lombana, N., Pérez Holguín, W. (2010). Uso de la robótica educativa como herramienta en los procesos de enseñanza *GIRA* 10 (1), 15 – 23.

Raposo, M.; Fuentes, E. y González, M. (2006). Desarrollo de competencias tecnológicas en la formación inicial de maestros. *Revista Latinoamericana de Tecnología Educativa* 5 (2), 525-537.

Román-Graván, P., Hervás-Gómez, C., y Guisado-Lizar, J. L. (2017). Experiencia de innovación educativa con robótica en la Facultad de Ciencias de la Educación de la Universidad de Sevilla (España). Recuperado el 07 de septiembre de 2020 de: <https://idus.us.es/handle/11441/65614>

Rosas, M.V., Zúñiga, M.E., Fernández, J.M., y Guerrero, R.A. (2017). El Pensamiento Computacional: experiencia de su aplicación en el aprendizaje de la resolución de problemas. XXIII Congreso Argentino de Ciencias de la Computación. Recuperado el 10 de marzo de 2019 de <http://sedici.unlp.edu.ar/handle/10915/63918>

Ruiz Ortega, F. (2007). Modelos Didácticos para la enseñanza de las Ciencias Naturales. *Revista Latinoamericana de Estudios Educativos*, 3 (2), 41-60.

Sabino, C. (2000). *Cómo hacer una tesis*. Buenos Aires: Lumen.

Sanz, C., Madoz, C., Zangara, A. y Albanesi, B. (s.f). El trabajo colaborativo y cooperativo mediado por TICs. Herramientas informáticas utilizadas en la mediación y experiencias realizadas. Recuperado el 10 de septiembre de 2020 de: http://sedici.unlp.edu.ar/bitstream/handle/10915/21995/Documento_completo.%20Herramientas%20informaticas%20utilizad.pdf?sequence=1&isAllowed=y

Scherer, F. (2018, 16 de septiembre). Generación Z. La hora del relevo. *La Nación*. Recuperado el 19 de enero de 2019 de <https://www.lanacion.com.ar/lifestyle/generacion-z-hora-del-relevo-nid2171319>

Tecnología, programación y robótica en secundaria. Usos Educativos de la Robótica. (2016). Recuperado el 20 de noviembre de 2016, de <https://tecnopujol.wordpress.com/2016/01/20/usos-educativos-de-la-robotica/>

La Robótica Educativa es un sistema de enseñanza interdisciplinaria que potencia el desarrollo de habilidades y competencias en los alumnos. (s.f). Recuperado el 26 de noviembre de 2016 de <http://www.edukative.es/que-es-la-robotica-educativa/>

Ventura, A.C., Gagliardi; R. y Moscoloni, N.A. (2012). Estudio descriptivo de los Estilos de Aprendizaje de estudiantes universitarios argentinos. *Revista Estilos de Aprendizaje*, 9 (5), 71-84.

Willing, P.A., Astudillo, G.J., Castro, L., Bast, S., Ocelli, M. Distel. J. (s.f). Educación con Tecnologías: la Robótica Educativa Aplicada para el Aprendizaje de la Programación. Recuperado el 6 de diciembre de 2018 de <http://sedici.unlp.edu.ar/handle/10915/62859>

Zapata-Ros, M. (15/09/2015). Pensamiento computacional: Una nueva alfabetización digital. Recuperado el 11 de junio de 2019 de: <https://www.um.es/ead/red/46/zapata.pdf>

SITIOS WEB

<https://www.buscojobs.com.ar/>

<https://www.computrabajo.com.ar/>

<https://www.zonajobs.com.ar>

<http://www.bumeran.com.ar/>

http://el.media.mit.edu/logo-foundation/what_is_logo/logo_primer.html

ANEXO I Encuesta a estudiantes – año 2017

Datos personales

1. Edad (años cumplidos): -----
2. Lugar de origen: -----
3. Lugar de Residencia :-----
4. Distancia (en Km) de su lugar de residencia a la UNNOBA:-----

Aspecto Laboral

5. ¿Cuántas horas trabaja a diario?

Trabajo más de 8 horas	Trabajo entre 7 y 8 horas	Trabajo entre 4 y 6 horas	Trabajo menos de 4 horas	No trabajo

Otras actividades

6. En cuanto al óseo y las actividades recreativas, semanalmente:

Le dedico más de 8 horas	Le dedico entre 7 y 8 horas	Le dedico entre 4 y 6 horas	Le dedico menos de 4 horas	No realizo actividades de óseo y recreativas.

Ámbito del hogar

7. En su hogar vive:

Solo	Con padres y hermanos	Con amigos	Con compañeros de la universidad	Con familia propia (pareja y/o hijos)

8. En su hogar ¿cuenta con espacio físico dónde estudiar?

Si	Ocasionalmente	Nunca

9. Cuando estudia IPI, lo hace:

Siempre solo	Siempre con otros compañeros	Ocasionalmente sólo y otras con compañeros

10. Fuera de los horarios de clase ¿Cuántas horas le dedica a IPI semanalmente?

Menos de 3 horas	Entre 3 y 4 horas	Entre 4 y 5 horas	Entre 5 y 6 horas	Más de 6 horas

Conocimientos previos de programación de computadoras

11. Antes de comenzar la universidad ¿sabía de qué se trataba la tarea de programar?

Si	Parcialmente	No

12. Para el caso en el que haya respondido positivamente a la respuesta anterior, ¿dónde aprendió a programar?

En la escuela	En Internet	Le enseñó un amigo	Una combinación de las opciones anteriores

En cuanto a la asignatura

13. ¿Cuánto puede decir que comprende las explicaciones la docente de teoría?

Totalmente	Parcialmente	Ocasionalmente	Entiendo muy poco	No comprendo nada en absoluto

14. Evaluando los materiales de la teoría ¿qué tan entendibles son?

Completamente entendibles	Poco entendibles	Muy poco entendibles	Nada entendibles

15. ¿Cuánto puede decir que comprende las explicaciones la docente de práctica?

Totalmente	Parcialmente	Ocasionalmente	Entiendo muy poco	No comprendo nada en absoluto

16. Los ejercicios trabajados en las clases prácticas ¿qué tan fáciles de resolver son?

Muy fáciles	Parcialmente fáciles	Difíciles	Muy Difíciles	Imposibles de resolver

17. Los tiempos que se establecen para resolver los diferentes trabajos prácticos le resultan:

Suficientes	Poco suficientes	Insuficientes

18. En cuanto a las evaluaciones le resultan:

Muy fáciles	Parcialmente fáciles	Difíciles	Muy Difíciles

19. Para aprobar las evaluaciones ¿Le parece que con lo trabajado en clases...

es suficiente?	es parcialmente suficiente?	es insuficiente?

20. Si tuvo dificultades dentro de la asignatura, ¿cómo lo califica en cuanto a cada uno de los siguientes ítems?

Ítems	Tiene relación directa	Tiene relación parcial	No tiene relación en lo absoluto
¿Considera que se debe a los tiempos dispuestos por las docentes?			
¿Tiene relación con el grado de dificultad de los temas vistos en IPI?			
¿Se relaciona con el modo en el que se trabaja en la teoría?			
¿Se relaciona con el modo en el que se trabaja en la práctica?			
¿Tiene que ver con el diseño de los enunciados de las evaluaciones?			

¿Qué tanto se debe a la preparación que traía de la escuela secundaria?			
¿Tiene relación con el tiempo que le dedica a IPI en general?			
¿Se relaciona con el interés que le presenta IPI?			
Tiene que ver con el grado de motivación que le despiertan los materiales de IPI.			

ANEXO II Formulario de Google – año 2018

Introducción a la Programación Imperativa

Todas las preguntas son opcionales. Si querés recibir una respuesta, dejanos tu e-mail al final del formulario

¿Tenés dificultad con algún o algunos temas de la asignatura? En ese caso, ¿cuáles?

Texto de respuesta larga

¿Comprendés los conceptos que explica la docente de teoría?

1 2 3 4 5

Se entiende muy poco

Se entiende todo

¿Comprendés los conceptos que explica la docente de práctica?

2 3 4 5

Se entiende muy poco

Se entiende todo

¿Qué actividades te resultan más útiles durante las clases? (marcá todas las que consideres)

Explicaciones de las docentes de los temas

Explicaciones de las docentes sobre ejercicios prácticos

Tiempo que la docente nos deja para trabajar solos

Consultar dudas de ejercicios que hice en mi casa

¿Qué sugerencia le harías a las docentes (de teoría o práctica) para que te resulten más útiles sus clases?

Texto de respuesta larga

¿Considerás que podés llevar la asignatura al día?

Sí

Parcialmente

No

Si no podés llevar la asignatura al día, ¿cuál o cuáles te parecen que son las causas?

Texto de respuesta corta

Si necesitás ayuda con un ejercicio concreto, escribí acá tu pregunta

Texto de respuesta larga

Con respecto al material teórico de la asignatura ¿te resulta accesible?

- Si
- Parcialmente
- No

Con respecto a las prácticas de la asignatura ¿te resulta posible resolverlas con lo que ves en clase?

- Si
- Parcialmente
- No

¿En qué comisión estás?

- JUNÍN - Teoría: miércoles 8hs / Práctica: miércoles 17hs
- JUNÍN - Teoría: miércoles 8hs / Práctica: martes 9hs
- JUNÍN - Teoría: miércoles 8hs / Práctica: martes 18hs
- PERGAMINO - Teoría: jueves 18hs / Práctica: martes 8hs
- PERGAMINO - Teoría: jueves 18hs / Práctica: jueves 8hs

Si querés, dejanos tu nombre

Texto de respuesta corta

Si querés que te respondamos, dejanos tu e-mail

Texto de respuesta corta

Si tuvieras algún otro comentario que quisieras realizar por favor escribirlo acá

Texto de respuesta larga

ANEXO III Trabajo práctico con ejercicios de REP

EJERCICIOS A RESOLVER CON REP

Ejercicios

1. **Elaborar un ejercicio para indicarle al REP que recorra dos cuadras por la calle 1, luego una cuadra por la calle 6.**

Objetivo: que el estudiante comience a probar sentencias en un problema sencillo de resolver, introduciéndose de este modo al uso de la herramienta.

Se presentan dos versiones que pueden llegar a ser soluciones que surjan en el aula.

```
ALGORITMO, versión 1
  Avanzar()
  Avanzar()
  Girar a la derecha()
  Avanzar()
```

```
ALGORITMO, versión 2
  Repetir 2-veces
    Avanzar()
  Girar a la derecha()
  Avanzar()
```

Nota: la primera versión es la esperada, la segunda es probable que surja luego de abordar el concepto estructura de control.

2. **Elaborar un ejercicio para indicarle al REP que recorra dos cuadras por la calle 1 y luego una cuadra por la calle 6, pero esta vez para el caso de encontrar algún obstáculo debe detenerse.**

Objetivo: que el estudiante comience a probar sentencias en un problema sencillo de resolver, introduciéndose de este modo al uso de la herramienta.

Se presentan dos versiones que pueden llegar a ser soluciones que surjan en el aula.

```
ALGORITMO, versión 1
  Avanzar()
  Si ¿Hay obstáculo?
    Detenerse()
  Avanzar()
  Si ¿Hay obstáculo?
    Detenerse()
  Si no ¿Hay obstáculo?
    Girar a la derecha()
  Avanzar()
```

```
ALGORITMO, versión 2
  Repetir 2-veces
    Avanzar()
    Si ¿Hay obstáculo?
      Detenerse()
  Si no ¿Hay obstáculo?
```

```
Girar a la derecha()  
Avanzar()
```

Nota: en este ejercicio particularmente se fuerza a trabajar con la estructura de control “Alternativa condicional”, ya que deben utilizar el sensor; sin embargo nuevamente la primera versión es la esperada, mientras que la segunda versión probablemente surja más adelante al abordar la estructura de control repetición simple.

3. Elaborar un ejercicio para indicarle al REP que recorra dos cuadras por la calle 1, luego una cuadra por la calle 6, y que luego regrese por donde vino.

Objetivo: que el estudiante continúe probando sentencias en un problema sencillo de resolver, pero agregándole una pequeña dificultad.

Se presentan dos versiones que pueden llegar a ser soluciones que surjan en el aula.

```
ALGORITMO, versión 1  
  Avanzar()  
  Avanzar()  
  Girar a la derecha()  
  Avanzar()  
  Girar a la derecha()  
  Girar a la derecha()  
  Avanzar()  
  Girar a la izquierda()  
  Avanzar()  
  Avanzar()
```

```
ALGORITMO, versión 2  
  Repetir 2-veces  
    Avanzar()  
  Girar a la derecha()  
  Avanzar()  
  Repetir 2-veces  
    Girar a la derecha()  
  Avanzar()  
  Girar a la izquierda()  
  Repetir 2-veces  
    Avanzar()
```

4. Elaborar un ejercicio para indicarle al REP que recorra dos cuadras por la calle 1, luego una cuadra por la calle 6. Si hubiera obstáculo en la esquina de las calles 1 y 6, REP debe regresar al punto de partida; caso contrario debe finalizar el recorrido.

Objetivo: que el estudiante vaya afianzando sus habilidades para utilizar la estructura de control “Alternativa condicional”.

Se presentan dos versiones que pueden llegar a ser soluciones que surjan en el aula.

```

ALGORITMO, versión 1
  Avanzar()
  Avanzar()
  Si ¿Hay obstáculo?
    Detenerse()
    Girar a la derecha()
    Girar a la derecha()
    Avanzar()
    Avanzar()
  Sino
    Girar a la derecha()
    Avanzar()

```

```

ALGORITMO, versión 2
  Repetir 2-veces
    Avanzar()
  Si ¿Hay obstáculo?
    Detenerse()
    Repetir 2-veces
      Girar a la derecha()
    Repetir 2-veces
      Avanzar()
  Sino
    Girar a la derecha()
    Avanzar()

```

5. Elaborar un ejercicio para indicarle al REP que recorra el perímetro del cuadrado que se forma al con las manzanas: M1, M2, M5 y M4; por las calles: 1, 6, 5 y 2.

Objetivo: introducir al estudiante a la estructura de control “Repetición simple”.

Se presentan cuatro versiones que pueden llegar a ser soluciones que surjan en el aula.

```

ALGORITMO, versión 1
  Avanzar()
  Avanzar()
  Girar a la derecha()
  Avanzar()
  Avanzar()
  Girar a la derecha()
  Avanzar()
  Avanzar()
  Girar a la derecha()
  Avanzar()
  Avanzar()

```

```

ALGORITMO, versión 2
  Repetir 4-veces
    Avanzar()
    Avanzar()
    Girar a la derecha()

```

```

ALGORITMO, versión 3

```

```

Repetir 4-veces
  Repetir 2-veces
    Avanzar()
  Girar a la derecha()

```

```

ALGORITMO, versión 4
  Repetir 3-veces
    Repetir 2-veces
      Avanzar()
    Girar a la derecha()
  Repetir 2-veces
    Avanzar()

```

Nota: en este punto la “versión 1” no debería plantearse, pero para los casos en los que sí sucediera el docente debería inducir al estudiante para que reconozca patrones en el algoritmo sugiriéndole sutilmente que piense en alguna solución en donde se utilice una estructura de control adecuada. La segunda versión utiliza la estructura “Repetición simple” pero aun así se puede mejorar, tal como se muestra en la tercera y última versión.

6. Elaborar un ejercicio para indicarle al REP que recorra el perímetro del cuadrado que se forma al con las manzanas: M1, M2, M5 y M4; por las calles: 1, 6, 5 y 2. Y que verifique si en cada uno de los extremos del cuadrado que se formó (es decir en las esquinas de las calles: 1 y 6, 5 y 6, 2 y 5) hay obstáculos, para esos casos REP debe detenerse.

Objetivo: continúan trabajando en la introducción del estudiante en la estructura de control “Repetición simple” y combinarla con “Alternativa simple” y utilización del sensor.

Se presentan dos versiones que pueden llegar a ser soluciones que surjan en el aula.

```

ALGORITMO, versión 1
  Avanzar()
  Avanzar()
  Si ¿Hay obstáculo?
    Detenerse()
  Sino
    Girar a la derecha()
    Avanzar()
    Avanzar()
    Si ¿Hay obstáculo?
      Detenerse()
    Sino
      Girar a la derecha()
      Avanzar()
      Avanzar()
      Si ¿Hay obstáculo?
        Detenerse()
      Sino
        Girar a la derecha()
        Avanzar()
        Avanzar()

```

```

ALGORITMO, versión 2

```

```

Repetir 2-veces
  Avanzar()
Si ¿Hay obstáculo?
  Detenerse()
Sino
  Girar a la derecha()
  Repetir 2-veces
    Avanzar()
  Si ¿Hay obstáculo?
    Detenerse()
  Sino
    Girar a la derecha()
    Repetir 2-veces
      Avanzar()
    Si ¿Hay obstáculo?
      Detenerse()
    Sino
      Girar a la derecha()
      Repetir 2-veces
        Avanzar()

```

7. Elaborar un ejercicio para indicarle al REP que recorra el perímetro del rectángulo que se forma al recorrer las manzanas: M1 y M2; por las calles: 1, 6, 3 y 2.

Objetivo: continúan trabajando en la introducción del estudiante en la estructura de control “Repetición simple”.

Se presentan tres versiones que pueden llegar a ser soluciones que surjan en el aula.

```

ALGORITMO, versión 1
  Repetir 2-veces
    Avanzar()
    Avanzar()
    Girar a la derecha()
    Avanzar()
    Girar a la derecha()

```

```

ALGORITMO, versión 2
  Repetir 2-veces
    Repetir 2-veces
      Avanzar()
    Girar a la derecha()
    Avanzar()
    Girar a la derecha()

```

```

ALGORITMO, versión 3
  Repetir 2-veces
    Avanzar()
  Girar a la derecha()
  Avanzar()
  Girar a la derecha()
  Repetir 2-veces
    Avanzar()
  Girar a la derecha()
  Avanzar()

```

8. Elaborar un ejercicio para indicarle al REP que recorra el camino en forma de escalera dado por las manzanas: 1, 5 y 9; recorriendo las calles: 1, 4, 3, 6, 5 y 8.

Objetivo: continúan trabajando en la introducción del estudiante en la estructura de control “Repetición simple” e introducir el concepto de función.

Se presentan cuatro versiones que pueden llegar a ser soluciones que surjan en el aula.

```
ALGORITMO, versión 1
  Avanzar()
  Girar a la derecha()
  Avanzar()
  Girar a la izquierda()
  Avanzar()
  Girar a la derecha()
  Avanzar()
  Girar a la izquierda()
  Avanzar()
  Girar a la derecha()
  Avanzar()

ALGORITMO, versión 2
  Repetir 3-veces
    Avanzar()
    Girar a la derecha()
    Avanzar()
    Girar a la izquierda()

ALGORITMO, versión 3
  Repetir 2-veces
    Avanzar()
    Girar a la derecha()
    Avanzar()
    Girar a la izquierda()
  Avanzar()
  Girar a la derecha()
  Avanzar()

ALGORITMO, versión 4
  Repetir 2-veces
    Recorrer un escalón ()
  Avanzar()
  Girar a la derecha()
  Avanzar()

Recorrer un escalón ()
  Avanzar()
  Girar a la derecha()
  Avanzar()
  Girar a la izquierda()
```

Nota: la sentencia Recorrer un escalón () es un llamado a una función.

9. Elaborar un ejercicio para indicarle al REP que recorra el camino en forma de escalera dado por las manzanas: 1, 5 y 9; recorriendo las calles: 1, 4, 3, 6, 5 y 8. Luego si hubiese obstáculo en la esquina de las calles 8 y 7, debe regresar.

Objetivo: continúan trabajando en la introducción del estudiante en la estructura de control “Repetición simple”, así como también el uso de sensores y “Alternativa condicional”.

Se presentan cuatro versiones que pueden llegar a ser soluciones que surjan en el aula.

```
ALGORITMO, versión 1
  Avanzar()
  Girar a la derecha()
  Avanzar()
  Girar a la izquierda()
  Avanzar()
  Girar a la derecha()
  Avanzar()
  Girar a la izquierda()
  Avanzar()
  Girar a la derecha()
  Avanzar()
  Si ¿Hay obstáculo?
    Girar a la derecha()
    Girar a la derecha()
    Avanzar()
    Girar a la izquierda()
    Avanzar()
    Girar a la derecha()
    Avanzar()
    Girar a la izquierda()
    Avanzar()
    Girar a la derecha()
    Avanzar()
    Girar a la izquierda()
    Avanzar()

ALGORITMO, versión 2
  Repetir 3-veces
    Avanzar()
    Girar a la derecha()
    Avanzar()
    Girar a la izquierda()
  Si ¿Hay obstáculo?
    Girar a la izquierda()
    Repetir 3-veces
      Avanzar()
      Girar a la izquierda()
      Avanzar()
      Girar a la derecha()

ALGORITMO, versión 3
  Repetir 2-veces
    Avanzar()
    Girar a la derecha()
```

```

    Avanzar()
    Girar a la izquierda()
Avanzar()
Girar a la derecha()
Avanzar()
Si ¿Hay obstáculo?
    Repetir 2-veces
        Girar a la derecha()
Repetir 2-veces
    Avanzar()
    Girar a la izquierda()
    Avanzar()
    Girar a la derecha()
Avanzar()
Girar a la izquierda()
Avanzar()

```

```

ALGORITMO, versión 4
    Repetir 2-veces
        Recorrer un escalón hacia abajo ()
Avanzar()
Girar a la derecha()
Avanzar()
Si ¿Hay obstáculo?
    Repetir 2-veces
        Girar a la derecha()
    Repetir 2-veces
        Recorrer un escalón hacia arriba
        ()
Avanzar()
Girar a la izquierda()
Avanzar()

```

```

Recorrer un escalón hacia abajo ()
    Avanzar()
    Girar a la derecha()
    Avanzar()
    Girar a la izquierda()

```

```

Recorrer un escalón hacia arriba ()
    Avanzar()
    Girar a la izquierda()
    Avanzar()
    Girar a la derecha()

```

Nota: las sentencias *Recorrer un escalón hacia abajo ()* y *Recorrer un escalón hacia arriba ()* son llamados a funciones.

10. Elaborar un ejercicio para indicarle al REP que REP recorra el perímetro del rectángulo que se forma al recorrer las manzanas: M1, M2, M5, M8, M7 y M4; por las calles: 1, 6, 7 y 2. Escribir soluciones que utilicen funciones.

Objetivo: continuar trabajando en la introducción del estudiante en la estructura de control “Repetición simple”, así como también el uso de funciones.

Se presentan tres versiones que pueden llegar a ser soluciones que surjan en el aula.

```
ALGORITMO, versión 1
  Repetir 2-veces
    Avanzar()
  Girar a la derecha()
  Repetir 3-veces
    Avanzar()
  Girar a la derecha()
  Repetir 2-veces
    Avanzar()
  Girar a la derecha()
  Repetir 3-veces
    Avanzar()

ALGORITMO, versión 2
  Repetir 2-veces
    Repetir 2-veces
      Avanzar()
    Girar a la derecha()
  Repetir 3-veces
    Avanzar()
  Girar a la derecha()

ALGORITMO, versión 3
  Repetir 2-veces
    Repetir 2-veces
      Avanzar()
    Dibujar base ()
    Girar a la derecha()
  Dibujar altura ()
  Repetir 3-veces
    Avanzar()
  Girar a la derecha()

Dibujar base ()
  Repetir 2-veces
    Avanzar()

Dibujar altura ()
  Repetir 3-veces
    Avanzar()
```

Nota: las sentencias *Dibujar base ()* y *Dibujar altura ()* son llamados a funciones.

11. Elaborar un ejercicio para indicarle al REP que recorra el perímetro de la ciudad a menos que encuentre un obstáculo que le impida continuar. Los obstáculos pueden estar en las esquinas de las calles 1 y 8, 8 y 7, 7 y 2; pero vamos a suponer que en la esquina de las calles 1 y 2 no hay obstáculo.

Objetivo: continuar trabajando en la introducción del estudiante en la estructura de control “Repetición simple”, así como también el uso de funciones y “Alternativa condicional”.

Se presentan tres versiones que pueden llegar a ser soluciones que surjan en el aula.

```
ALGORITMO, versión 1
  Repetir 3-veces
    Avanzar()
  Si ¿Hay obstáculo?
    Detenerse()
  Sino
    Girar a la derecha()
    Repetir 3-veces
      Avanzar()
    Si ¿Hay obstáculo?
      Detenerse()
    Sino
      Girar a la derecha()
      Repetir 3-veces
        Avanzar()
      Si ¿Hay obstáculo?
        Detenerse()
    Sino
      Girar a la derecha()
      Repetir 3-veces
        Avanzar()

ALGORITMO, versión 2
  Recorrer un lado del cuadrado ()
  Si ¿Hay obstáculo?
    Detenerse()
  Sino
    Girar a la derecha()
    Recorrer un lado del cuadrado ()
    Si ¿Hay obstáculo?
      Detenerse()
    Sino
      Girar a la derecha()
      Recorrer un lado del cuadrado ()
      Si ¿Hay obstáculo?
        Detenerse()
    Sino
      Girar a la derecha()
      Recorrer un lado del
      cuadrado ()

  Recorrer un lado del cuadrado
    Repetir 3-veces
      Avanzar()
```

Nota: la sentencia *Recorrer un lado del cuadrado ()* es un llamado a la función.