

ProgEnv: Un entorno para la enseñanza de la programación

Arroyo Marcelo, Bavera Francisco, Bengolea Valeria, Cerda Ernesto, Kilmurray Cecilia, Novaira Maria Marta, Permigiani Sonia, Ponzio Pablo, Scilingo Gastón, Tardivo María Laura, Uva Marcelo

Universidad Nacional de Río Cuarto,
Río Cuarto, Argentina

{marcelo.arroyo, pancho, vbengolea, ecerda, ckilmurray, mnovaira, spermigiani, gaston, lauratardivo, uva}@dc.exa.unrc.edu.ar

Abstract. El curso de Ingreso Universitario en carreras de computación, es para muchos estudiantes, el primer contacto con la disciplina y por ello es de gran importancia el desarrollo de actividades que permitan introducir los conceptos básicos, favoreciendo la participación de los estudiantes a través de consignas que estimulen la creatividad y el interés por las ciencias de la computación, en particular por la programación. La complejidad de las actividades abordadas es sin duda un factor determinante para promover el aprendizaje de conceptos asociados a la programación, siendo muchas veces motivo de frustración para alumnos que se inician en esta práctica. En este contexto, la elección de las herramientas y lenguajes de enseñanza adecuados juegan un rol fundamental, permitiendo desarrollar nociones complejas con un alto grado de abstracción de manera simple y clara. En este trabajo presentaremos un entorno de desarrollo visual para la enseñanza de la programación, al que llamamos *ProgEnv* por *Programming Environment*, su utilización en el dictado del módulo “Introducción a la Programación” que forma parte de las actividades de ingreso a las carreras Analista en Computación, Profesorado y Licenciatura en Ciencias de la Computación de la Facultad de Ciencias Exactas de la Universidad Nacional de Río Cuarto y los resultados obtenidos en dicha experiencia.

1 Introducción

La enseñanza de la programación tanto en niveles educativos iniciales como en cursos introductorios universitarios, es objeto constante de estudio en el mundo. Muchos lenguajes y entornos de programación han sido desarrollados con el propósito de introducir nociones asociadas a la programación [10, 5, 1, 2, 7], en los cuales se propone representar las instrucciones con bloques que se pueden encastrar y desencastrar para formar programas minimizando de esta manera, problemas de tipo sintáctico, los cuales suelen ser muy frustrantes para aquellos que se inician en la programación. De este modo, se simplifica la tarea de construcción de programas poniendo el foco en la creatividad y el ingenio para resolver problemas algorítmicos. Asociado a estos lenguajes, se han desarrollado y

están disponibles para su uso, muchas plataformas online [4, 9, 6], que proveen secuencias de actividades utilizando como base este tipo de lenguajes para enseñar nociones tales como repetición, condicional, abstracción procedural, etc.

Este tipo de lenguajes y plataformas están siendo utilizadas en muchas instituciones educativas, en particular el Departamento de Computación de la facultad de Ciencias Exactas Físico Químicas y Naturales de la Universidad Nacional de Río Cuarto ha utilizado durante los últimos años una variedad de estos lenguajes y plataformas en el dictado del módulo Introducción a la Programación que forma parte de la currícula de ingreso a las carreras: Analista en Computación, Profesorado y Licenciatura en Ciencias de la Computación. El programa curricular de este módulo consiste básicamente en análisis de problemas, diseño de soluciones algorítmicas, lenguajes y conceptos introductorios de programación; consta de 7 clases de 3 horas cada una y persigue como propósito principal brindar una primera mirada de la disciplina general, y en particular de la programación, pensada desde una perspectiva vocacional que despierte entusiasmo por las Ciencias de la Computación en los estudiantes. Por otra parte, también se utilizan en actividades orientadas a estudiantes de los últimos años del ciclo escolar anterior. Estas actividades tienen una clara intención vocacional.

Los lenguajes de programación visuales han resultado propicios para el dictado del módulo anteriormente descrito por varias razones: proveen una interfaz intuitiva para los alumnos permitiendo la construcción de programas simples en la primera clase, ayudan a estimular la creatividad y entrenar a los estudiantes en la resolución de problemas algorítmicos, sin dejar de mencionar la posibilidad de utilizar una innumerable cantidad de actividades que se encuentran disponibles como parte de las plataformas. En contraparte, motivaron el desarrollo de este trabajo algunas dificultades encontradas en el sucesivo dictado del módulo en años anteriores:

- Los alumnos presentan dificultad en trasladar los conceptos aprendidos utilizando los lenguajes visuales a un lenguaje textual. Creemos que esto se debe a que, debido al escaso tiempo de duración del módulo, no es posible poner énfasis en este punto.
- Los docentes del módulo pretenden realizar un pequeño taller que permita introducir los conceptos básicos de programación de manera incremental a través de una secuencia de ejercicios cuya composición culmine en el desarrollo de un juego simple. El principal objetivo de este taller es captar el interés de los alumnos, abordando consignas desafiantes que le permita a los alumnos percibir la aplicación de los nociones enseñadas. Muchas de las plataformas existentes no proveen actividades configurables sino una secuencia de ejercicios que poseen un hilo temático en común.

En este artículo presentamos *ProgEnv* [8], un ambiente de programación web para la enseñanza de la programación. El mismo consiste en un entorno gráfico mediante el cual es posible utilizar recursos funcionales, predefinidos por el docente, permitiendo la incorporación progresiva de elementos sintácticos y construcciones lógicas del lenguaje de programación, en este caso particular JavaScript [14], aunque podría extenderse para incorporar el uso de otros

lenguajes de programación. Además, *ProgEnv* le provee al docente una interfaz de programación flexible para la incorporación y modificación de actividades disponibles de manera sencilla e intuitiva. A través de la experiencia en el uso de *ProgEnv* en el dictado del módulo Introducción a la Programación del ingreso a las carreras de computación de la Universidad Nacional de Río Cuarto en febrero de 2019 pudimos observar una significativa mejora en los conceptos adquiridos por los estudiantes al finalizar el dictado del módulo, donde la mayoría de los estudiantes logró los objetivos planteados. En las secciones a continuación describiremos en detalle el ambiente desarrollado, las funcionalidades que posee, las ventajas observadas durante su uso, algunas actividades abordadas utilizando el entorno y los resultados obtenidos en tal experiencia.

2 *ProgEnv*: Entorno de Desarrollo para la enseñanza de la programación

ProgEnv es un ambiente de desarrollo web pensado para ser utilizado en cursos introductorios de programación, su nombre tiene origen en el término en Inglés: *Programming Environment*. La herramienta fue desarrollada en HTML y JavaScript utilizando *w2ui* [13] como librería para la creación de la interfaz de usuario y *codeMirror* [3], aportando una experiencia gráfica placentera a los alumnos y docentes. El entorno puede ser utilizado usando sólo un navegador web y está disponible en la web de ingresantes del Departamento de Computación [11]. Éste puede ser utilizado tanto dentro del aula como fuera. Cabe mencionar que utilizar un navegador web como ambiente de desarrollo trae consigo algunos beneficios:

- No requiere la instalación por parte de los usuarios de ningún software extra.
- El docente puede integrar, de forma natural, recursos multimedia como imágenes o videos, así como también vincular la aplicación con la API de otros sitios web
- Todos los actores, alumnos y docentes, están familiarizados con su uso.

Al igual que la mayoría de las plataformas de enseñanza basadas en lenguajes de programación con bloques, *ProgEnv* dispone de un área para escribir el programa y otra para visualizar los resultados de su ejecución. Esta última se compone de un lienzo para exhibir resultados gráficos y de una consola para mostrar resultados que comprenden texto. El entorno incluye un visualizador de la posición del mouse cuando éste es arrastrado por encima del lienzo. Esta funcionalidad, puede ser usada en la resolución de problemas que requieren conocer la ubicación exacta de un “click” sobre el lienzo, o utilizar el lienzo como un plano de coordenadas cartesiana, por ejemplo para dibujar figuras geométricas en posiciones o tamaños específicos del mismo.

El entorno cuenta con una barra lateral izquierda donde se muestran las actividades disponibles para el alumno. La selección de una actividad específica, permitirá, en el área inferior de la pantalla, visualizar las instrucciones disponibles

para lograr los objetivos planteados en dicha actividad, así como también el enunciado de la actividad misma. Las instrucciones disponibles están predefinidas por el docente como bloques modulares (no visibles al alumno) en el lenguaje JavaScript. Con el objetivo de esclarecer lo antedicho, supongamos una actividad sencilla:

Arrojar un dado, si el número obtenido es par entonces dibujar un triángulo, caso contrario, dibujar un círculo.

Asumimos que se cuenta con las siguientes instrucciones (que pre-define el docente y se despliegan al seleccionar la actividad):

- `tirarDado()`: Devuelve de manera aleatoria un número entre 1 y 6, y escribe el número aleatorio en la consola.
- `esPar(nro)`: Retorna `true` si `nro` es par, y `false` en caso contrario
- `dibujarCirculo(ox,oy, r)`: Dibuja un círculo con centro (ox,oy) y radio r .
- `dibujarTriangulo(ox,oy, l)`: Dibuja un triángulo equilátero con vértice izquierdo (ox,oy) y lado de longitud l .
- `cambiarAnchoLapiz(nro)`: Cambia el ancho del lápiz de dibujo al ancho dado por `nro`
- `cambiarColorLapiz(c)`: Cambia el color del lápiz de dibujo al color c . colores disponibles: *verde, rojo, blanco, negro, azul, amarillo y violeta*.

El conjunto de instrucciones predefinidas podrá ser utilizado en conjunto con las estructuras de control (`if then else` en este caso) para construir una solución al problema planteado. Mediante este conjunto de instrucciones predefinidas se define la capa de abstracción que el docente pretende, según los conceptos que desea exponer en la actividad. Por ejemplo, el docente podría no brindar la instrucción `esPar(nro)` como parte de la interfaz de programación, y proponer que el alumno implemente esta función. Como otro ejemplo, si se tratase de una actividad inicial, el docente podría proporcionar instrucciones `dibujarCirculo()` y `dibujarTriangulo()`, que dibujan respectivas figuras de algún tamaño y en una posición fija del lienzo, en lugar de proveer instrucciones genéricas que requieren argumentos, para luego, moverse a un conjunto de instrucciones más generales en etapas más avanzadas del dictado del módulo.

Es importante notar que, a diferencia de las plataformas basadas en bloques el alumno podrá utilizar, además de la API predefinida, las estructuras de control, abstracciones, definición de variables, etc del lenguaje JavaScript (Aunque si fuera de interés, el docente también podría definir abstracciones para tales construcciones).

En la figura 1, se muestra una solución del problema descripto.

El entorno resulta adecuado para definir una secuencia de actividades como la del ejemplo, que mantengan un hilo en común, para alcanzar de manera incremental, reusando creativamente lo realizado, algún objetivo general.

Algunos puntos que diferencian *ProgEnv* de otros entornos para la enseñanza de la programación:



Fig. 1. Solución al problema enunciado en la sección 2

- Permite que un estudiante que recién se inicia en la programación pueda escribir un programa en un lenguaje textual (JavaScript en este caso) utilizando el conjunto de instrucciones propuesta por el docente.
- Posibilita el desarrollo de actividades de manera incremental y en diferentes niveles de abstracción, atendiendo la diversidad y heterogeneidad de los ingresantes. Esta característica posibilita su uso en otros niveles educativos.
- El docente es quien define, según los objetivos de la clase, la API de la que dispondrá el alumno para resolver las actividades de la clase, dando la posibilidad al docente de generar propuestas creativas que despierten el entusiasmo de los estudiantes.
- Es simple de utilizar dentro y fuera del aula ya que no requiere más que de un navegador web. Inclusive desde un celular el estudiante puede realizar las actividades propuestas.

2.1 JavaScript como lenguaje de programación inicial

Como se ha mencionado en las secciones anteriores de este trabajo, *ProgEnv* soporta JavaScript como lenguaje de programación.

Si bien la elección de JavaScript como lenguaje base tiene su fundamento principal en cuestiones técnicas: *ProgEnv* es una aplicación del lado del cliente, y en este contexto ejecutar un programa JavaScript es trivial; es importante mencionar algunas consideraciones que han hecho adecuada esta elección en el ámbito del módulo dictado:

- En la práctica sólo hemos usado un subconjunto del lenguaje, el cual incluye:
 - Composición secuencial
 - Declaración y uso de variables (**Var**)
 - Asignación simple: =

- Operadores de comparación: `==`, `!=`, `>`, `<`, etc
- Operadores lógicos : *and* (`&&`), *or* (`||`) y *not* (`!`)
- Operadores aritméticos estándar: suma (+), resta (−), multiplicación (*), división (/) y resto (%)
- Constantes String: ej: “hola mundo”
- Constantes booleanas: `true` y `false`
- Estructuras de control:
 - * `if then`
 - * `if then else`
 - * `for(inicialización; condición; incremento)`
 - * `while(condición)`
- Abstracción procedural : funciones con y sin parámetros. Funciones que retornan valores (`return`)

La selección de este subconjunto del lenguaje se basó en las necesidades específicas del curso y las actividades abordadas, de acuerdo al escaso período de duración del ingreso: 7 clases introductorias.

- El subconjunto de JavaScript mencionado resulta suficiente para abordar actividades cuyo objetivo principal es iniciar a los estudiantes en la resolución algorítmica de problemas captando el interés y despertando la creatividad y el ingenio de los mismos. Teniendo en cuenta este punto, no es necesario introducir conceptos más avanzados del lenguaje (objetos, eventos, etc).
- El docente puede utilizar otras características específicas del lenguaje ocultando detalles a los estudiantes.
- Si bien javascript es un lenguaje con tipado dinámico, la introducción de tipos de datos no fue un objetivo del curso de ingreso, por lo que esto no resultó, en nuestro caso, un inconveniente. Cabe destacar que la mayoría de las plataformas mencionadas en la introducción de este trabajo, no abordan en primera instancia tipo de datos. Por otro lado, es bien conocido el lenguaje TypeScript [12], un superconjunto de JavaScript que esencialmente añade tipado estático, por lo que si fuera necesario *ProgEnv* podría extenderse para soportar este lenguaje.

3 Ingreso Universitario 2019: Una experiencia de uso

La intención de esta sección es presentar alguna de las actividades llevadas a cabo utilizando *ProgEnv* durante el dictado del módulo “Introducción a la Programación” que forma parte del ingreso universitario 2019.

En esta experiencia, las actividades desarrolladas utilizando *ProgEnv* permitieron abordar, de manera incremental, conceptos básicos de programación estimulando razonamientos lógico-matemáticos necesarios para resolver situaciones problemáticas interesantes.

Las primeras actividades fueron diseñadas y realizadas con el objetivo de introducir el entorno de desarrollo y nociones iniciales de programación. Se plantearon actividades tales como dibujar figuras geométricas simples empleando las instrucciones predefinidas (presentadas como un ejemplo en la sección

2), donde además de ir conociendo el entorno de desarrollo, los estudiantes aplicaron conceptos de composición secuencial, condicional (ver figura 1); y operadores lógicos y matemáticos de manera muy natural. Estas actividades motivaron a aquellos estudiantes curiosos y creativos a escribir diferentes programas para realizar dibujos más complejos.

En una de las actividades propuestas se presentó el Lienzo con dos líneas perpendiculares ya dibujadas (horizontal y vertical) quedando éste dividido en cuatro cuadrantes iguales. La tarea consistió en escribir un programa que identifique e informe el cuadrante en el que se produjo un *click*. Para realizar esto se proporcionó el siguiente conjunto de instrucciones:

- `escribir(mensaje)`: Muestra el mensaje dado en la Consola
- `obtenerX()`: devuelve la coordenada X del punto (dentro del lienzo) en donde se hizo *click*
- `obtenerY()`: devuelve la coordenada Y del punto (dentro del lienzo) en donde se hizo *click*

A diferencia del ejemplo dado en la sección anterior, en este caso la ejecución del programa ocurre cuando se hace *click* en el Lienzo. Cabe destacar que, nuevamente las instrucciones predefinidas ocultan los detalles del manejo de eventos.

Esta como otras actividades abordadas, tuvieron como objetivo proporcionar conocimientos y herramientas que ayuden a dibujar y manipular un tablero de *TaTeTi*, actividad que se presentó llegando al final del módulo.

El desarrollo del *TaTeTi*, consistió en dibujar el tablero (conformado por una grilla de 3x3), identificar la posición dentro de la grilla en donde se produjo un *click* y, según el jugador (Rojo o Azul) de turno, dibujar en la posición de la grilla *clikeada* un cuadrado rojo o un cuadrado azul. Cabe destacar que, la posibilidad de predefinir el conjunto de instrucciones, y proporcionar variables y funciones ya definidas en el área de código (por ejemplo, asociadas al cambio de turno), ayudó significativamente a que los estudiantes pudieran completar esta actividad. La secuencia de pasos completa para llegar al desarrollo de este juego se puede ver en la página web de Ingresantes del Departamento de Computación [11].

Por último, el desarrollo de una función que pudiera identificar el jugador ganador también fue, de manera guiada por el grupo de docentes, implementada por los estudiantes.

Es importante destacar que la mayoría de los alumnos pudo llegar al final del desarrollo, los alumnos no solo se mostraron muy entusiasmados en realizar esta actividad sino también en compartir con el resto del grupo los logros alcanzados.

Una evaluación diagnóstico final permite valorar las actividades realizadas en el módulo. En la misma se pudo observar que un 70% de los estudiantes evaluados lograron adquirir los conceptos abordados, tales como, lenguaje, diseño de soluciones algorítmicas, conceptos básicos de programación, estructuras de control y abstracción, siendo éste un resultado altamente alentador para continuar con el camino emprendido en esta experiencia.

4 Trabajos a Futuro

Como hemos mencionado en diversas ocasiones en este artículo, *ProgEnv* fue utilizado en una única experiencia: el ingreso universitario 2019 a las carreras de computación de la Universidad Nacional de Río Cuarto. De dicha experiencia han surgido varias propuestas para darle continuidad a este proyecto. En lo que se refiere al entorno en sí mismo, se propone desarrollar un primer prototipo cliente-servidor que cuente con las siguientes características:

- Posibilidad de crear y administrar cuentas de usuarios con vista para docentes y alumnos, permitiendo de esta manera que los alumnos pueden crear soluciones a las actividades planteadas, guardarlas y modificarlas. La vista del docente, permitirá crear y manipular actividades con una interfaz de usuario integrada que permita de manera simple codificar instrucciones pre-definidas y ocultar detalles no relevantes para el alumno.
- Soporte a diferentes lenguajes de programación

Por último, desde el punto de vista de la enseñanza, se pretende estudiar otras experiencias similares de enseñanza para evaluar intensivamente tanto la elección del lenguaje de programación como las actividades propuestas.

5 Conclusiones

En este artículo hemos presentado *ProgEnv* un ambiente de programación web para la enseñanza de la programación. El desarrollo de este ambiente surge de la necesidad de repensar las actividades de ingreso a las carreras Analista en Computación, Profesorado y Licenciatura en Ciencias de la Computación de la Facultad de Ciencias Exactas de la Universidad Nacional de Río Cuarto. La utilización de *ProgEnv* durante el dictado del módulo Introducción a la Programación actuó como hilo conductor en las diferentes clases planificadas, logrando una incorporación progresiva de los conceptos abordados. Las actividades desarrolladas utilizando *ProgEnv* nos permitieron introducir de manera incremental los distintos conceptos, estructuras de control y abstracciones culminando en la implementación del juego TaTeTI.

Es importante remarcar que los estudiantes se mostraron motivados en cada una de las actividades propuestas y como se destacó en la sección 2, el examen diagnóstico final arrojó que un 70% de los estudiantes evaluados lograron adquirir los conceptos abordados.

Por último, el desarrollo y uso de *ProgEnv* brindó la oportunidad de incorporar las valiosas experiencias de docentes que han estado involucrados en numerosas oportunidades en las actividades de ingreso y pre-ingreso facilitando por otra parte, el inicio de los estudiantes a las primeras materias específicas de la disciplina.

References

1. Alice: <https://www.alice.org/>.
2. Appinventor: <https://appinventor.mit.edu/>.
3. Codemirror: <https://codemirror.net/>.
4. Code.org: <https://code.org/>.
5. Gobstones: <https://gobstones.github.io/>.
6. Mumuki: <https://mumuki.io/paths>.
7. Pilas bloques: <http://pilasbloques.program.ar/>.
8. Progenv: <https://github.com/marcelodarroyo/progenv>.
9. Program.ar: <http://program.ar/>.
10. Scratch: <https://scratch.mit.edu/>.
11. Sitio del Departamento de Computación, UNRC: <https://dc.exa.unrc.edu.ar>.
12. Typescript: <https://www.typescriptlang.org/>.
13. w2ui: <http://w2ui.com/web/>.
14. Axel Rauschmayer. *Speaking JavaScript: An In-Depth Guide for Programmers*. O'Reilly Media; 1 edition, 2014.