

# ESTRUCTURA Y HERRAMIENTAS PARA UN CURSO INICIAL SOBRE PROGRAMACION ORIENTADA A OBJETOS

Carlos Lombardi ([carlos.lombardi@unq.edu.ar](mailto:carlos.lombardi@unq.edu.ar)) - UNQ / UNAHUR

Nicolás Passerini ([npasserini@gmail.com](mailto:npasserini@gmail.com)) - UNQ / UNSAM

## RESUMEN

Presentamos un proyecto sobre enseñanza de la Programación Orientada a Objetos (POO), en el que se estudian tanto las posibles estructuraciones de un curso inicial sobre POO, como las herramientas a utilizar en el mismo. Partimos de las dificultades que los docentes asociados al proyecto notamos en el dictado de cursos desde hace más de 10 años, y de la literatura existente sobre el tema. Se formalizó una estructura de curso en cuatro etapas, y se desarrolló un lenguaje y entorno de programación didácticos que acompañan la secuencia sugerida. El entorno incluye facilidades avanzadas de edición, una consola interactiva, soporte para tests de ejecución automática, generación automática de diagramas, errores de codificación adecuados a estudiantes iniciales que aparecen mientras se tipea, entre otras características.

Estos resultados se utilizan en cursos de 5 universidades nacionales desde 2015. También hemos presentado nuestros resultados en eventos nacionales e internacionales.

Actualmente estamos trabajando en distintas direcciones con el objetivo de que otros docentes puedan incorporar los resultados de nuestro trabajo.

Hay un tesista de doctorado, uno de licenciatura, y dos becarios estudiantes colaborando en el proyecto.

*Palabras clave: programación orientada a objetos, enseñanza de programación, entornos educativos, lenguajes de programación educativos.*

## CONTEXTO

Este documento describe un trabajo de investigación aplicada, cuyo campo de estudio es la enseñanza inicial de la POO. Los objetos de estudio son las posibles formas de estructurar un curso, y los entornos de programación de propósito educativo adecuados.

Participan docentes pertenecientes a varias universidades, habiéndose generado varios proyectos de investigación. Actualmente está vigente el proyecto PUNQ EXPTE. 1373/17 "Entornos y herramientas de desarrollo orientados a objetos" de la Universidad Nacional de Quilmes. También estuvieron relacionados con el tema proyectos de las Universidades Tecnológica Nacional y Nacional de San Martín, ya concluidos.

Los resultados de este trabajo en curso son aprovechados para el dictado de carreras de informática en la Universidad Tecnológica Nacional (regionales Buenos Aires y Delta), y las Universidades Nacionales de Quilmes, San Martín, Avellaneda y Hurlingham. En todas ellas se utiliza el entorno/lenguaje Wollok, cuyo desarrollo es parte integral del trabajo.

## 1. INTRODUCCION

El marco de este trabajo son los cursos iniciales sobre POO de nivel universitario, asumiendo que los estudiantes han cursado previamente una materia de programación.

Varios de los docentes que participan en este proyecto llevan más de 10 años dictando cursos iniciales sobre POO en distintas universidades.

En el aprendizaje de los conceptos y técnicas asociados a la POO, confluyen dificultades notadas en la generalidad de los cursos iniciales

de programación, con otras relacionadas específicamente con la POO.

En la experiencia de los docentes que participan en este proyecto, hemos notado en particular:

- la dificultad en el manejo de expresiones complejas, p.ej. el encadenamiento de mensajes o la inclusión de expresiones en los parámetros de una invocación.
- confusiones en el uso una interfaz de objetos para trabajar con listas o colecciones (uso de map, filter, reduce, etc.).
- muy significativamente, problemas para diferenciar los conceptos de clase e instancia, en particular dificultad para comprender la instanciación.
- falta de capacidad para aplicar el concepto de polimorfismo en la resolución de problemas.

El objetivo general del proyecto es propender al dictado de cursos iniciales de POO en los que las dificultades encontradas se mitiguen en la medida de lo posible, y se traten específicamente cuando no se encuentren estrategias adecuadas de mitigación.

Distinguimos dos objetos de estudio que consideramos igualmente relevantes para el logro de los objetivos del proyecto.

El primero está conformado por el planteo y la organización de un curso. En particular, trabajamos sobre el conjunto de conceptos, ideas y técnicas conviene transmitir y practicar en un curso inicial sobre POO; las etapas en las que conviene estructurar un curso; las posibles secuencias didácticas que se derivan de las estructuración propuesta; y la generación de material de apoyo para docentes y estudiantes.

El segundo es el conjunto de herramientas de programación a utilizar en un curso inicial sobre POO. Trabajamos bajo la hipótesis de la conveniencia de utilizar un entorno integrado de desarrollo (IDE, por las siglas en inglés) y un lenguaje de programación concebidos especialmente para su uso en este marco. Hemos desarrollado sucesivamente varias

propuestas de entornos y/o lenguajes, que se realimentan a partir de la experiencia obtenida en los cursos que están a cargo de docentes relacionados con el proyecto.

Los dos objetos de estudio están estrechamente relacionados. Por caso, en la propuesta de estructuración en cuatro etapas que formalizamos recientemente, cada etapa está relacionada con características y construcciones específicas de un lenguaje de programación de propósito didáctico que hemos definido en el marco del proyecto.

Nuestro trabajo se ha nutrido de varios trabajos presentes en la literatura, relacionados con los objetos de estudio de este proyecto.

Respecto del planteo y la organización de un curso, en [Mey03] y [KQPR03] se proponen enfoques progresivos respecto de las tareas que realiza un estudiante a lo largo del curso. Por otro lado, en [BC04] se sugiere basar un curso sobre POO en la construcción de modelos de complejidad progresivamente creciente, en donde el trabajo con los conceptos básicos se da en espiral de acuerdo al patrón definido en [Ber00]. Finalmente, en [CO16] se propone introducir gradualmente los elementos que forman un lenguaje de programación. Las propuestas elaboradas en este proyecto retoman características de todas estas propuestas.

Respecto de los lenguajes y entornos de programación, mencionemos el debate sobre el uso de lenguajes de programación educativos o industriales, cfr. [ACS02], [PS+07].

Dentro de los primeros, existen varias propuestas, algunas para cursos iniciales de programación [MBS12,Pat81] y otras referidas específicamente a POO [BR+96]. En particular el lenguaje Blue [KR96] tiene varios puntos de contacto con Wollok, lenguaje desarrollado en el marco del proyecto.

También hay varios trabajos específicos sobre entornos de programación educativos para POO. Varios de ellos, en particular Traffic [Mey03],

BlueJ [KQPR03] y Dr.Java [ACS02], comparten características con los desarrollados durante este proyecto, p.ej. la inclusión de tests de evaluación automática, un intérprete interactivo para la experimentación rápida, o el cuidado en los mensajes de error que se presentan al estudiante.

También tenemos en cuenta las motivaciones que llevaron a la definición de lenguajes educativos basados en bloques [CDP00,ML07] o en frames [Köl16,MOM15], en particular la posibilidad de utilizar herramientas gráficas en el aprendizaje de la programación.

## 2. LÍNEAS DE INVESTIGACIÓN Y DESARROLLO

Definimos dos líneas de trabajo, relacionadas con cada uno de los objetos de estudio que describimos más arriba.

Por un lado, nos proponemos estudiar y definir estructuras y secuencias pedagógicas para un primer curso de POO, suponiendo estudiantes que realizaron previamente un curso básico de programación. Teniendo en cuenta las dificultades encontradas en cursos dictados en años anteriores, que mencionamos en la introducción, concluimos que debe ponerse especial cuidado en los siguientes aspectos:

- introducción de la noción de polimorfismo en la etapa más temprana posible.
- manejo cuidadoso de los conceptos de clase e instanciación.
- trabajo con expresiones, aumentando la complejidad de las expresiones que se utilizan a medida que avanza el curso.

También nos proponemos estudiar en qué punto de un curso inicial sobre POO, y de qué forma, introducir nociones ligadas al desarrollo de software, que no pertenecen al ámbito estricto de la programación. En particular, estamos interesados en testeos automatizados, y en el uso de repositorios de código.

Finalmente, consideramos como un aspecto relevante en nuestros estudios y definiciones, la articulación del primer curso de POO con los siguientes cursos de programación, y la capacidad de aprovechar efectivamente los conocimientos adquiridos en la continuación del trayecto formativo de cada estudiante, y posteriormente en el ámbito profesional.

En la segunda línea de trabajo, investigamos sobre qué características deberían buscarse en un lenguaje y un entorno de programación, para que faciliten el desarrollo de un curso organizado de acuerdo a las secuencias pedagógicas que proponemos.

En particular, evaluamos la conveniencia de utilizar un lenguaje industrial o uno de propósito educativo, y con qué herramientas y características debería contar un entorno de programación para estudiantes.

Durante el desarrollo de esta investigación se generaron varios entornos de programación con propósito educativo. Sucesivamente, surgieron Object Browser, Loop y Ozono [GL+11], todos ellos basados en el lenguaje Smalltalk.

Algunas limitaciones que comprobamos en el uso de estos entornos en cursos que dictamos, nos llevaron a definir un lenguaje de propósito educativo, y acompañarlo con un entorno integrado de desarrollo, también orientado al uso por estudiantes. Así surgió Wollok [JPT14,NL+17,Wol], un producto que combina lenguaje y entorno, usado desde 2015 en varias universidades. Hasta la fecha, Wollok ha sido utilizado en más de 40 cursos universitarios, con un total aproximado de 1500 estudiantes.

Dado que docentes asociados al proyecto dictan cursos de POO en varias universidades, podemos testear continuamente los resultados que vamos obteniendo en el trabajo de investigación, tanto respecto de secuencias pedagógicas, como de herramientas de programación a utilizarse. Esto genera que los resultados de nuestras investigaciones tienen un impacto concreto en la enseñanza de programación en varias universidades, y

además, una realimentación que resulta indispensable en la continuidad del trabajo de investigación y desarrollo.

### **3. RESULTADOS OBTENIDOS/ESPERADOS**

Respecto del trabajo en organización de un curso, hemos podido formalizar una propuesta de estructuración en cuatro etapas o estadios, que se corresponden con subconjuntos crecientes del lenguaje Wollok.

Las transiciones, tanto de una etapa a la siguiente como considerando el curso como un todo, son objeto de una atención especial.

Entre los trabajos en curso mencionamos:

- sistematización de la base de enunciados de ejercicios, seleccionando enunciados adecuados para cada etapa, y vinculando cada enunciado con los conceptos que se pretende ejercitar específicamente. El objetivo es generar material de guía que facilite la difusión y utilización de los resultados de este trabajo.
- integración con la plataforma Mumuki, que refuerza la posibilidad de difusión, y permite recolectar fácilmente información estadística acerca de la performance de los estudiantes.
- sistematización del análisis cualitativo de resoluciones de ejercicios de modelado utilizando POO, que permitan extraer conclusiones acerca de fortalezas y debilidades de variantes en la organización de un curso. Se individualizaron más de 15 aspectos que pueden estudiarse, y se está preparando el material necesario para llevar a cabo estudios sobre grandes cantidades de estudiantes.

Respecto de las herramientas a utilizarse, tanto los resultados alcanzados como los desarrollos en curso se refieren al lenguaje y entorno Wollok [JPT14,NL+17], y a Wollok-game, una extensión del mismo que facilita la generación de juegos interactivos.

La base de la definición del lenguaje Wollok es estable desde hace varios años, y se ha utilizado exitosamente en varias universidades. El

lenguaje admite su introducción en subconjuntos crecientes, que se corresponden con las etapas de la organización de curso que proponemos. Además, su sintaxis permite una transición sencilla a los lenguajes industriales Java y Javascript.

El entorno Wollok es utilizado desde 2015, en más de 40 cursos. Cuenta con varias funcionalidades deseables para un curso inicial sobre POO, entre las que destacamos

- editor con coloreo de sintaxis, autocompletado e indicación de errores mientras se tipea.
- una consola interactiva REPL (Read-evaluation-print loop) que permite experimentar con las características del lenguaje y los objetos y clases desarrollados por el estudiante.
- mensajes de error trabajados para su comprensión por parte de los estudiantes.
- detección de varios defectos de codificación comunes en estudiantes iniciales.
- inclusión de una herramienta para definir y ejecutar tests de evaluación automática.

En 2018 se incorporó Wollok-game, que se utilizó para desarrollar juegos gráficos interactivos en clase.

Entre los trabajos en curso, mencionamos

- el desarrollo de una versión de Wollok que pueda usarse desde un browser Web, evitando la necesidad de instalación.
- la inclusión de una interfaz simplificada con repositorios de código Git.
- varios agregados a Wollok-game, entre ellos soporte de sonido, mejoramiento de las interfaces de código, y un mejor trabajo respecto de las colisiones entre objetos con representación gráfica.

### **4. FORMACIÓN DE RECURSOS HUMANOS**

Colaboran activamente en este proyecto docentes de las universidades en las que se utiliza Wollok, en distintos roles: recolectando y analizando la información sobre el uso de nuestras propuestas y herramientas en el dictado

de materias, formalizando la propuesta se secuencia pedagógica, o generando material para docentes y estudiantes.

Nicolás Passerini está desarrollando su trabajo de Doctorado de Ciencia y Tecnología en la UNQ sobre esta temática. La tesis de Licenciatura en Informática en la UNQ de Ronny de Jesús consiste en un prototipo de entorno Wollok que pueda utilizarse desde un navegador Web. El trabajo de integración profesional para la Tecnicatura en Programación Informática UNQ de Lucas Sivestri consiste en el desarrollo de una interfaz gráfica de Git a ser integrada dentro de Wollok. El director de estos trabajos es Carlos Lombardi, quien también es director del proyecto de investigación vigente en la UNQ.

Federico Aloï, quien está cursando estudios de profesorado, trabaja como becario de este proyecto, en la sistematización del material para alumnos y la integración de Wollok con la plataforma Mumuki.

## 5. BIBLIOGRAFIA

- [ACS02] E.Allen, B.Cartwright, B.Stoler: "DrJava: a lightweight pedagogic environment for Java", ACM SIGCSE Bulletin 34, 2002.
- [BC04] J.Bennedsen, M.E.Caspersen: "Teaching object-oriented programming - towards teaching a systematic programming process". In ECOOP 2004.
- [Ber00] J.Bergin: "Fourteen pedagogical patterns". In EuroPLoP 2000.
- [BR+96] J.Bergin, J.Roberts, R.Pattis, M.Stehlik: Karel++: a gentle introduction to the art of object-oriented programming. Wiley, EE.UU, 1996.
- [CDP00] S.Cooper, W.Dann, R.Pausch: "Alice: a 3-d tool for introductory programming concepts". Journal of Computing Sciences in Colleges 15, 2000.
- [CO16] W.Cazzola, D.M.Olivares, "Gradually learning programming supported by a growable programming language". IEEE Transactions on Emerging Topics in Computing 4(3), 2016.
- [FPT14] J.Fernandes, N.Passerini, P.Tesone: "Wollok - relearning how to teach object-oriented programming". In Workshop de Ingeniería de Software y Tecnologías Informáticas, UTN, 2014.
- [GL+11] C.Griggio, G.Leiva, G.Polito, G.Decuzzi, N.Passerini: "A programming environment supporting a prototype-based introduction to OOP". In IWST 2011.
- [Köl16] M.Köllings, "Lessons from the design of three educational programming environments: Blue, BlueJ and Greenfoot.". International Journal of People-Oriented Programming 4, 2016.
- [KQPR03] M.Köllings, B.Quig, A.Patterson, J.Rosenberg: "The BlueJ system and its pedagogy". Computer Science Education, 13(4), 2003.
- [KR96] M.Köllings, J.Rosenberg: "Blue - a language for teaching object-oriented programming". In SIGCSE 1996.
- [MBS12] P.Martínez López, E.Bonelli, F.Sawady "El nombre verdadero de la programación", 2012.
- [Mey03] B.Meyer: "The outside-in method of teaching introductory programming". In PSI 2003.
- [ML07] D.Malan, H.Leitner: "Scratch for building computer scientists". SIGCSE Bulletin 39(1), 2007.
- [MOM15] J.Monig, Y.Ohshima, J.Maloney: "Blocks at your fingertips: blurring the line between blocks and text in gp", in Blocks and Beyond Workshop, IEEE 2015.
- [Pat81] R.Pattis: Karel the Robot: a gentle introduction to the art of programming. Wiley, EE.UU, 1981.
- [PL+17] N.Passerini, C.Lombardi, J.Fernandes, P.Tesone, F.Dodino, "Wollok: Language + ide for a gentle and industry-aware introduction to OOP". In LACLO 2017.
- [PS+07] A.Pears, S.Seidman, L.Malmi, L.Mannilla, E.Adams, J.Bennedsen, M.Devlin, J.Paterson: "A survey of literature on the teaching of introductory programming". ACM SIGCSE Bulletin 39, 2007.
- [Wol] [www.wollok.org](http://www.wollok.org).