

Component-Oriented Development Applied to Strengthening of Logical Capabilities for Program

Jesús Reyes Carvajal Carvajal, M.Sc¹, Oscar Agudelo Varela, M.Sc², Juan Fajardo Barrera, Esp³,
Universidad de los Llanos¹²³, Colombia¹³², jreyes@unillanos.edu.co¹, oscar.agudelo@unillanos.edu.co²,
jfajardo@unillanos.edu.co³

Abstract– *This article describes the characteristics of an application that aims to strengthen or enhance the logic of students in computer programming courses. This tool is another contribution to the teaching strategies used by teachers in the teaching / learning of computer programming. With the use of open source libraries that facilitate the development of interactive environments the use and handling of components, led to the application called BLOCK-C, which allows the creation of algorithms based on logical assembly of components and similarly to visualize its corresponding C source code, this in order that students get to know the structure (syntax) of the C language and the way some instructions are encoded.*

Keywords- *Big Data, MapReduce, Hadoop, Schedulers, Execution Speculative*

Digital Object Identifier (DOI):<http://dx.doi.org/10.18687/LACCEI2018.1.1.331>
ISBN: 978-0-9993443-1-6
ISSN: 2414-6390

Desarrollo Orientado a Componentes Aplicado al Fortalecimiento de Capacidades Lógicas para Programar

Jesús Reyes Carvajal Carvajal, M.Sc¹, Oscar Agudelo Varela, M.Sc², Juan Fajardo Barrera, Esp³,
Universidad de los Llanos^{1,2,3}, Colombia^{1,2,3}, jreyes@unillanos.edu.co¹, oscar.agudelo@unillanos.edu.co²,
jfajardo@unillanos.edu.co³

Abstract– *This article describes the characteristics of an application that aims to strengthen or enhance the logic of students in computer programming courses. This tool is another contribution to the teaching strategies used by teachers in the teaching / learning of computer programming. With the use of open source libraries that facilitate the development of interactive environments the use and handling of components, led to the application called BLOCK-C, which allows the creation of algorithms based on logical assembly of components and similarly to visualize its corresponding C source code, this in order that students get to know the structure (syntax) of the C language and the way some instructions are encoded.*

Keywords– *Component Oriented development, libraries, open source.*

Resumen– *El presente artículo describe las características de una aplicación que tiene por objetivo fortalecer o mejorar la capacidad lógica de los estudiantes en los cursos de programación de computadores. Esta herramienta informática es un aporte más a las estrategias pedagógicas utilizadas por docentes en la enseñanza/aprendizaje de la programación de computadores. Con el uso de librerías de código abierto, que facilitan el desarrollo de entornos interactivos del uso y manejo de componentes, se dio origen a la aplicación denominada BLOCK-C, la cual permite la creación de algoritmos con base en el ensamble lógico de componentes y de igual forma permite visualizar su correspondiente código fuente en lenguaje C, esto con el fin que el estudiante vaya conociendo la estructura (sintaxis) del lenguaje C y la forma como se codifican algunas instrucciones.*

Palabras clave: *Desarrollo orientado a componentes, librerías, código abierto.*

I. INTRODUCCIÓN

Las dificultades en la enseñanza/aprendizaje de la programación han sido un problema recurrente en los últimos 20 años, tanto en nuestro país como en el mundo entero, la preocupación de la comunidad académica por estos cursos es, debido a la alta mortalidad en los cursos de programación, alta deserción, baja motivación de estudiantes, alto plagio y poco interés de los profesores por buscar nuevas estrategias de programación.

Se han propuesto algunas herramientas informáticas y metodologías pero ninguna proporciona una respuesta integral a la problemática. Eso sin contar con la aparición de otros enfoques y el surgimiento de nuevos lenguajes de

programación, herramientas y tecnologías, las cuales introducen nuevos factores de dificultad a la ya compleja labor de enseñar a programar.

En su artículo “El Aprendizaje Basado en Problemas. Experiencia Piloto en la Enseñanza de un Lenguaje de Programación” [1], las autoras realizaron una reflexión de la aplicación de una metodología, el Aprendizaje Basado en Problemas (ABP), la cual está funcionando satisfactoriamente en muchas universidades del mundo desde hace varios años. De la aplicación de esta metodología en un curso, por parte de un grupo de docentes, se obtuvieron los siguientes resultados. Los estudiantes se encuentran “muy satisfechos” por el desenvolvimiento del tutor 36%, muy cercano al 29%, opinaron que el “tutor cumplió de la mejor manera con su tarea”. También coincidieron, casi por mayoría, que la estructuración del tema así como el problema planteado sirvió, para cumplir con los objetivos que se perseguían, pero la escasez de tiempo disponible fue un obstáculo por no haber aplicado la metodología en los cuatro cursos de un cuatrimestre [1].

La Universidad de los Andes de Colombia, desde hace varios años, está utilizando una herramienta pedagógica llamada Cupi2. La cual cambió la forma en que aprenden a programar los estudiantes de los Andes. Cupi2 articula modelo pedagógico y de evaluación, materiales de soporte al aprendizaje, mecanismos de seguimiento y formación de profesores. Es referencia para cambiar los currículos de Ingeniería de Sistemas en universidades del país [2]. Los antecedentes que llevaron a aplicar esta herramienta fueron: “Hace cinco años, Programación I era uno de tres cursos problema entre todas las materias que se estudian en la Universidad de los Andes. Igual que en otros lugares del mundo en los últimos 20 años, los alumnos la odiaban, hasta el punto de que entre 30 y 35% reprobaba la materia e incluso hubo un semestre en el que esa ‘mortalidad’ alcanzó 75%. La desmotivación y la mala percepción se extendían a los otros dos cursos de Programación del currículo de Sistemas y Computación” [2]. Y los resultados después de haber aplicado esta herramienta fueron: “Hoy, gracias a Cupi2, una herramienta pedagógica apoyada en la tecnología y diseñada para flechar a los estudiantes igual que lo hace el travieso dios romano con los enamorados, la mortalidad es de 10-12%.

Incluso, hay muchachos de carreras como Música, Arte, Biología, Matemáticas, Física y Administración que la toman como electiva. Y ahora no la califican con 3,5 o 3,7 sobre 5, sino que obtiene notas entre 4,5 y 4,7” [2].

El Grupo de Tecnologías de la Información, de la Universidad Autónoma de Bucaramanga (UNAB), ejecuto el proyecto denominado “Desarrollo de prácticas para la enseñanza de la Programación en nivel introductorio, usando ALICE y Java” [3], este plantea el diseño de prácticas que integran el uso de Alice con el lenguaje Java, en los cursos de Fundamentos de Programación de la UNAB, busca afianzar los conceptos básicos de la programación orientada a objetos de una manera más amigable y atractiva.

Se cuenta con una gran cantidad de herramientas informáticas que tienen por objetivo enseñar a programar en diferentes entornos, se comentaran algunas, como es el caso de Scratch, lenguaje de programación que facilita crear sus propias historias interactivas, animaciones, juegos, música y arte, proyecto que fue lanzado desde el 2007 y fue desarrollado por el Media Lab del Massachusetts Institute of Technology (MIT). En su sitio web cuenta con ScratchED un apartado especial para educadores, en este se indican las posibilidades de usar Scratch en las aulas” [4]. En el año 2009 se creó ScratchEd, donde una comunidad en línea de docentes de todo el mundo comparte historias, intercambian recursos, hacen preguntas y se encuentran con otros educadores que usan Scratch.

Alice es un programa educativo, libre y abierto, orientado a objetos con un entorno de desarrollo integrado. Está programado en Java. Utiliza un entorno sencillo basado en arrastrar y soltar para crear animaciones mediante modelos 3D. Esto hace que sea mucho más avanzado que Scratch, algunas instituciones educativas primero comienzan con Scratch y luego utilizan Alice [5].

App Inventor para Android, “La aplicación Inventor utiliza bloques de construcción, de clases, para ayudarle a diseñar sus aplicaciones Android propias”. Al igual que Scratch App inventor tiene raíces en el MIT. En la Universidad de San Francisco, App Inventor se enseña como un curso del plan de estudios de las ciencias de la computación. Muchos de los estudiantes toman el curso, ya que detestan las matemáticas y el curso cubre este requisito temido [6].

En niveles de educación superior se espera de los estudiantes, que ciertas capacidades, como la lógica sean lo suficientemente bien definidas para abordar la carrera profesional que seleccionaron.

La capacidad lógica es inherente a nosotros y está bien formada, más en unos que en otros, pero la capacidad lógica para programar, que aunque no se aparta de la lógica, es más específica y exigente.

La sociedad de hoy se enfrenta al reto de incorporar nuevas tecnologías y más específicamente al uso del computador. Algunos lo utilizan como herramienta de trabajo, otros quieren aprender a programarlo, para crear aplicaciones o herramientas informáticas. Sin embargo se requiere que el programador de software tenga ciertas cualidades especiales para resolver problemas relacionados con la programación de computadores como: capacidad analítica para comprender los problemas que se le planteen, lógica y buen sentido común, capacidad de observación, espíritu autodidacta e investigativo y habilidad para estructurar una serie de pasos que conduzcan a la solución de los problemas tomando en cuenta las capacidades y limitantes de la computadora.

Los problemas que han tenido algunas instituciones académicas del país y del mundo, con respecto a la enseñanza/aprendizaje de los cursos de programación son solucionados generalmente por los docentes. Es habitual encontrar en los estudiantes de primer semestre problemas como: carencia de habilidades para programar, carencia de disciplina en programación, falta de conciencia estudiantil, desinterés por sus estudios en general y apatía por la materia en particular.

II. METODOLOGÍA

La aplicación se desarrolló bajo las directrices de la Técnica de Modelado de Objetos (OMT), ya que la aplicación controlaría componentes y en su fondo se pueden ver como objetos, permitiendo adaptarse a las necesidades actuales y futuras.

El desarrollo del proyecto utilizo la cadena de actividades ilustrada en la figura 1, apoyando la metodología OMT.

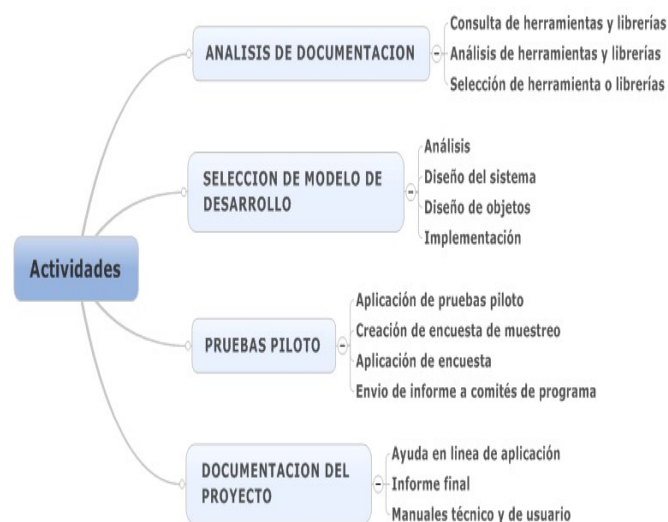


Fig 1. Cadena de actividades

Las fases de la metodología OMT desarrolladas en este proyecto son [7]:

A. Análisis

Teniendo claro el objetivo de lo que el sistema va hacer, se procede a desarrollar un modelo expresado en términos de objetos y de relaciones entre ellos, además un flujo dinámico de control y de transformaciones funcionales.

El modelo de objetos fue construido a partir de un diagrama de objetos, más los sitios donde se almacenan los datos del sistema (nombre, descripción, alias, contenido y organización), además se identifican los procesos que hacen uso de los datos con sus respectivas ubicaciones. El modelo dinámico fue creado con base en los diagramas de secuencia (diseñar algoritmo, generar código fuente, guardar algoritmo y actualizar algoritmo) y sus algunos diagramas de estado.

Por ultimo en las transformaciones funcionales se especificó el significado de los métodos y sus acciones, además se realizaron pruebas con los diagramas de flujo de datos para verificar si se obtienen los resultados esperados, el flujo del proceso de observa en la figura 2.

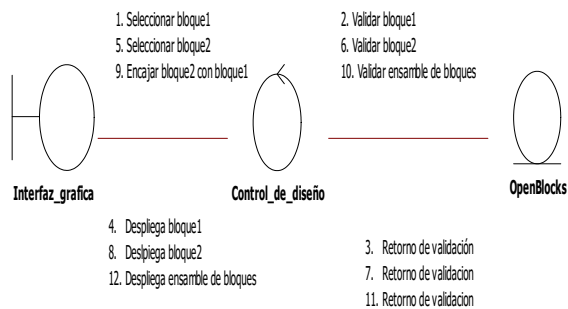


Fig 2. Diagrama de generación de algoritmo

B. Diseño del Sistema

Durante esta fase el sistema se organizó en subsistemas (niveles) basándose tanto en la estructura del análisis como en la arquitectura propuesta, ver figura 3.

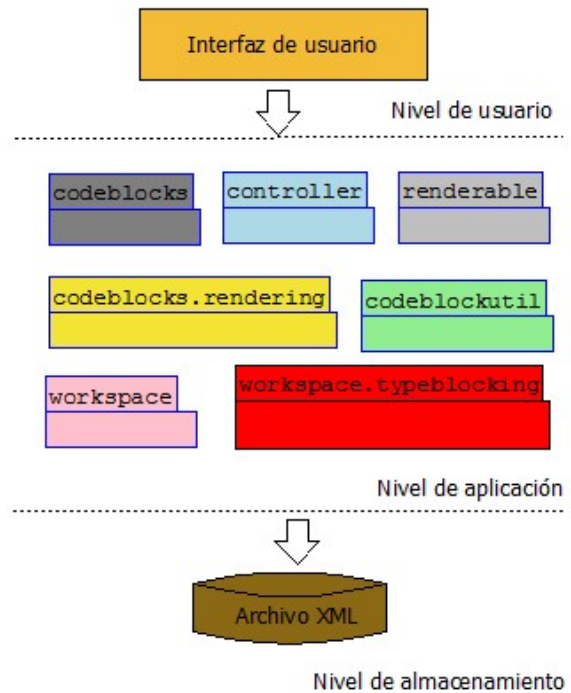


Fig 3. Diseño de sistema

C. Diseño de Objetos

Este diseño está enfocado en la relación de cinco clases, interfaz gráfica, control de diseño, generador de código, explorador de archivos y OpenBlocks. Esta última clase agrupa una serie de clases que hacen posible el uso y manejo de los componentes en un área de trabajo.

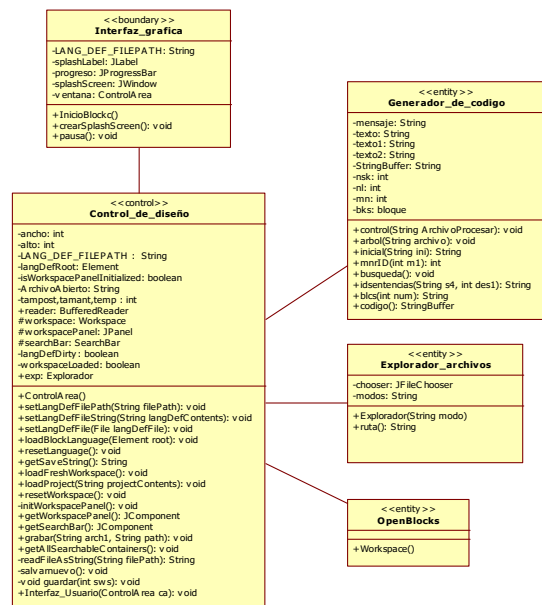


Fig 4. Diagrama de clases

D. Implementación

La aplicación fue desarrollada en Java, en el entorno de desarrollo integrado Eclipse (versión Indigo), sobre plataforma Linux (Ubuntu 10.04), permitiendo su ejecución sobre cualquier distribución Linux y Windows (Win7 y superiores).

Para conocer el grado de aceptación y cumplimiento de los objetivos propuestos para el desarrollo de la aplicación, se realizó una prueba piloto con los estudiantes de primer semestre de los programas de Ingeniería de Sistemas y Electrónica de la Universidad de los Llanos. En su primera fase se aplicó una prueba de rendimiento, en la cual se determinaron los tiempos promedio utilizados por los estudiantes para resolver cinco problemas de programación. Como una forma de evidenciar si la aplicación ofrecía los elementos básicos para dar solución a problemas de computación, el grupo de estudiantes por programa se dividió en dos grupos, donde un grupo utilizando la aplicación Block-C resolvería los cinco problemas y el otro grupo lo haría directamente en un compilador de C. En la segunda fase de la prueba piloto se aplicó una encuesta a los estudiantes que estuvieron en la primera fase. Esta encuesta tenía como objetivo conocer algunas inquietudes y opiniones de los estudiantes sobre la experiencia que tuvieron sobre el desarrollo de los problemas con la aplicación Block-C y sin ella.

III. RESULTADOS

La aplicación denominada Block-C tiene una interfaz gráfica amigable y de fácil manejo, la cual consta de una pantalla principal compuesta por tres secciones.

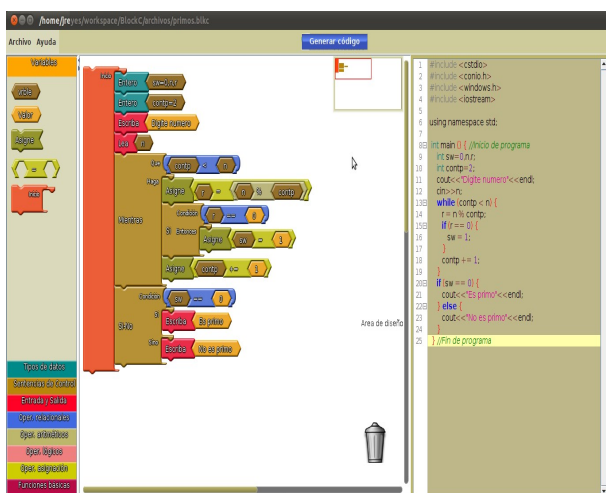


Fig 5. Aplicación Block-C

- *Cajón de bloques*

Ubicados en el lado izquierdo de la pantalla principal, donde se encuentra los bloques divididos por categorías (Definiciones, instrucciones de control, instrucciones de entrada y salida de datos, operadores relacionales, operadores aritméticos, operadores de asignación y funciones básicas) que representan los temas básicos del lenguaje C.

- *Área de diseño*

En esta área, el estudiante crea su algoritmo con base en el encaje lógico de bloques que selecciona del cajón de bloques, allí también se encuentra la papelerera de reciclaje de bloques.

- *Generación de código fuente*

En esta área aparece el código fuente en Lenguaje C, que la aplicación genera automáticamente con base en el algoritmo diseñado.

La aplicación tiene un menú principal, en el cual tiene las opciones de gestión de algoritmos (crear, modificar y guardar) y su correspondiente ayuda en línea.

El código fuente generado por la aplicación, en su defecto puede ser compilado en C++ o G++ y garantiza su ejecución libre de errores sintácticos, pero no lógicos. Además cualquier cambio realizado en el algoritmo es reflejado en el código fuente en el momento que el estudiante así lo decida.

De la prueba de rendimiento realizada con los estudiantes de primer semestre de ingeniería electrónica y de sistemas de la Universidad de los Llanos, se obtuvieron los siguientes resultados.

TABLA 1.
RESULTADOS DE PRUEBA PILOTO

	Muestra	Con Block-C	Sin Block-C
Electrónica	40	2047	2230
Sistemas	12	669	718

El tiempo total utilizado por los estudiantes de electrónica y sistemas para resolver cinco ejercicios con la aplicación Block-C, fueron menores al tiempo utilizado por estudiantes que no utilizaron la aplicación Block-C (los resolvieron directamente en un compilador de C).

IV. DISCUSIÓN

Respecto al desarrollo de la aplicación y pruebas de esta, con los estudiantes, se puede llegar a evidenciar que:

La gran mayoría de los 40 estudiantes encuestados, poco utilizan las aplicaciones informáticas para mejorar su capacidad lógica, dado que algunos docentes no utilizan o apoyan este tipo de estrategias didácticas para la enseñanza/aprendizaje de la programación de computadores.

El desarrollo de ejercicios por su cuenta (24 estudiantes) es la estrategia más utilizada para mejorar su capacidad lógica y como aplicación la más utilizada es Diagramas de Flujos.

En un alto porcentaje, si están de acuerdo que una aplicación informática puede mejorar su capacidad lógica.

La aplicación Block-C tiene los elementos básicos para dar solución a problemas de computación y si mejora la capacidad lógica de sus usuarios.

El código generado por la aplicación ofrece una mayor reducción de tiempo, respecto al utilizado por un programador cuando está codificando la solución de un problema directamente sobre un compilador de C.

Block-C fue pensada específicamente para fortalecer o despertar la capacidad lógica de los estudiantes a través de la creación de algoritmos, basándose en el ensamble lógico de componentes, muy diferente a las formas habituales de desarrollo de lógica como diagramas de flujo y pseudocódigo..

V. CONCLUSIONES

Block-C ofrece una estrategia nueva al diseño de algoritmos con base en el ensamble lógico de componentes, diferente a herramientas tradicionales como diagramas de flujo y pseudocódigo.

Con base en el código fuente generado automáticamente por la aplicación, el estudiante conocerá la sintaxis del lenguaje C y la forma como se estructuran instrucciones y sentencias de control, las cuales están representadas en componentes.

Hoy en día ya se encuentran librerías de código abierto que permiten la programación de componentes base y además tienen soporte para la manipulación de datos XML, como son OpenBlocks y JDOM respectivamente. Estas librerías son optimizadas para trabajar con Java y por tanto fueron el eje ideal para el desarrollo de la aplicación.

Técnicamente Block-C simula las funciones que realiza un Parsers, dado que el código fuente generado por la aplicación, es validado, para que esté bien estructurado y sea equivalente a la sintaxis del lenguaje C.

La aplicación Block-C tiene alto grado de aceptación por parte de los estudiantes, como lo confirma la prueba piloto que se realizó, dado que la aplicación ofrece una forma natural de expresar lógica para dar resolución a problemas de computador, además permite conocer su correspondiente código fuente en lenguaje C.

De la encuesta se aprecia que hay una gran cantidad de estudiantes que no conocen de estrategias metodológicas y aplicaciones informáticas que se pueden utilizar como apoyo a la enseñanza/aprendizaje de la programación de computadores.

REFERENCES

- [1] Martínez Fátima del C., and Torres Auad Lía. "El Aprendizaje Basado en Problemas Experiencia piloto en la enseñanza de un lenguaje de programación". Revista de ciencias exactas e ingeniería, 2010. Consultado el 2016 en: www.herrera.unt.edu.ar/revistacet/torresdocencia.pdf.
- [2] Villalobos Jorge, "Flechazos de Cupi2 para aprender a programar". Revista Contacto, numero 07, paginas 21-23, 2013.
- [3] Grupo de Tecnologías de la Información, GTI, Universidad Autónoma de Bucaramanga. 2012. Recuperado el 4 de agosto 2016, de www.unab.edu.co/portal/page/portal/UNAB/investigacion/investigacion-en-sentido-estricto/inicio/grupo/descripcion?idgrupo=1.
- [4] Migda Karen Chilpa Morga. "Las TIC en la Educación Aplicando las tecnologías en la educación". 2013, Recuperado <http://karenmijn.blogspot.com.co/2013/04/scratch-en-informatica-se-usa-en.html>
- [5] Carn Wanda Dann, Stephen Cooper, and Randy Pausch.. "Learning to Program with Alice, 3rd Edition". Pearson, 2012.
- [6] MIT Center for Mobile Learning. "App Inventor Beginner Tutorials". 2016, disponible en <http://appinventor.mit.edu/explore/sites/all/files/hourofcode/AppInventorTutorials.pdf>
- [7] Rajender Nath Vetter and Dharminder Kumar, "Introduction to Object-Oriented Methodology". 2009. Disponible en <http://www.ddegjust.ac.in/studymaterial/mca-5/mca-503.pdf>