

## Posibilidades y limitaciones para la introducción del pensamiento computacional en la educación básica

*Rodrigo Duda*  
*Universidade Tecnológica Federal do Paraná*  
[rodrigo.duda@utfpr.edu.br](mailto:rodrigo.duda@utfpr.edu.br)

*Sani de Carvalho Rutz da Silva*  
*Universidade Tecnológica Federal do Paraná*  
[sani@utfpr.edu.br](mailto:sani@utfpr.edu.br)

*Cleverson Sebastião dos Anjos*  
*Instituto Federal do Paraná – Campus Irati*  
[cleverson.anjos@ifpr.edu.br](mailto:cleverson.anjos@ifpr.edu.br)

*Roger Adriano Bressani Mazur*  
*Instituto Federal do Paraná – Campus Irati*  
[roger.mazur@ifpr.edu.br](mailto:roger.mazur@ifpr.edu.br)

*Fecha de recepción 14 de mayo 2016*

*Fecha de aceptación 12 de junio 2016*

### Resumen

En este trabajo, se discute sobre una propuesta didáctica que posibilite el abordaje del pensamiento computacional en las clases de matemáticas en la educación básica. Se exponen los presupuestos teóricos que amparan en la integración de una herramienta digital constructora en el proceso de construcción y organización del conocimiento matemático, con reflexiones sobre límites y posibilidades de esa práctica. Se remite al constructorismo, vertiente difundida por Papert, por creerse que la integración de una herramienta computacional en el proceso de enseñanza y aprendizaje es relevante si ésta realmente fomenta subsidios para un cambio significativo en la forma de pensar, expresarse y construir conocimiento.

**Palabras clave:** App Inventor, constructorismo, programación visual.

### Abstract

In this paper, we discuss about a didactic approach that enables the approach of computational thinking in math classes in basic education. The theoretical assumptions that protects the

integration of a constructionist digital tool in the process of construction and organization of mathematical knowledge, with reflections on limits and possibilities of this practice are discussed. It refers to constructionism, shed spread by Papert, because it was believed that the integration of a computational tool in the teaching and learning process is relevant if it really promotes subsidies for a significant change in the way we think, express and build knowledge.

**Keywords:** App Inventor, constructionism, visual programming.

## Introducción

El actual escenario educacional exige nuevas perspectivas para corresponder a los anhelos de una generación discente compuesta por individuos cada vez más ávidos por componentes tecnológicos. Porque están insertados en un contexto en que la tecnología digital es un elemento común en su cotidiano, Prensky (2001) clasificalos como nativos digitales. Mucho más de lo que apenas estén inseridos en un contexto tecnológico, Prensky destaca que esta generación discente procesa informaciones de una manera singular y característica.

El construccionismo, término acuñado por Papert, se refiere a la forma como el alumno produce conocimiento por medio de la interacción con la máquina. Seymour Papert, pionero en estudios sobre la temática, “considera que el ordenador debe permitir la construcción del conocimiento a través del aprender haciendo y del pensar sobre lo que se está haciendo”. (Campos, 2013, p. 69). Esa afirmación indica que el alumno debe ser sujeto activo en el proceso de enseñanza y aprendizaje, y no solo un mero receptor de conocimiento.

Es notable el deslumbramiento de los adolescentes de la actualidad con relación a recursos tecnológicos. En especial, los ordenadores y dispositivos móviles (*smartphones* y *tablets*) son elementos conocidos por una parte significativa de los estudiantes, siendo utilizados como medio de comunicación, entretenimiento y búsqueda de informaciones vía *web*.

Aunque no sea la solución de todos los problemas educacionales, es importante que la tecnología digital tenga espacio en el ambiente escolar. Un aspecto importante referente al uso del ordenador en el salón de clase, es el hecho de que él representa papel decisivo en la enseñanza de matemáticas, por posibilitar la construcción de modelos virtuales (Mendes, 2009, p. 113). Como herramienta de procesamiento de datos, posibilita el desarrollo de actividades que pueden

englobar distintos aspectos. Esas características son destacadas por Borba y Penteadó (2012), que recomiendan que las propuestas pedagógicas enfatizan la experimentación, visualización, simulación, comunicación electrónica y problemas abiertos.

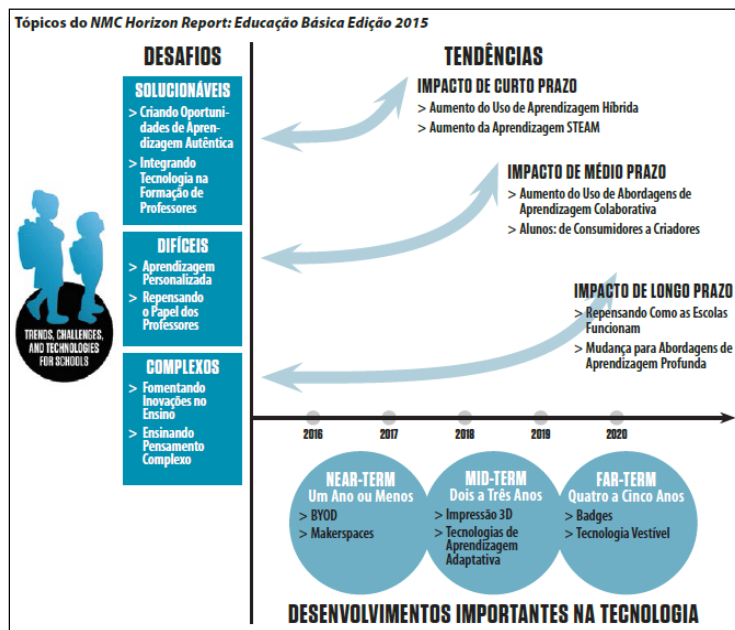
Para eso, es necesario que el docente reflexione sobre la necesidad real de insertar el ordenador en el salón de clase y si eso resultará en la mejoría de la calidad de las actividades desarrolladas (Cox, 2003), bajo el riesgo de que su uso no se relacione con objetivos y competencias que sean adquiridas por los alumnos y no sea efectivamente integrada al proceso de enseñanza, se forme en un simple aderezo. (Giraldo, Caetano & Mattos, 2013).

Considerando estos aspectos, en este trabajo se discurre sobre las posibilidades y limitaciones para la introducción del pensamiento computacional en la educación básica por medio de la utilización de una herramienta constructora de desarrollo de aplicaciones: el *App Inventor*.

### **El horizonte educacional para los próximos años y la posibilidad de insertar el pensamiento computacional en la educación básica**

El horizonte educacional de la educación básica presente en el documento “*NMC Horizon Report: Educação Educaçao Basica 2015*” contempla la adopción de tecnología educacional para promoción de aprendizaje colaborativo, aumento del uso de aprendizaje híbrido y aprendizaje STEAM (*Science, Technology, Engineering, Art and Math*), además de cambios para abordaje de aprendizaje profundo. Esas y otras características son clasificadas como posibles de implantación a corto, medio y largo plazo en los próximos 5 años, conforme ilustrado en la Figura 1.

**Figura 1:** Horizonte educacional con adopción de tecnologías previsto para los próximos 5 años



Fuente: NMC *Horizon Report*: Edición Educación Básica 2015

En el documento se presenta una visión consensual de 56 especialistas, convergiendo en una propuesta transformadora en la educación básica por medio de la adopción de tecnología educacional. Se discute sobre el impacto de la tecnología en el desarrollo de habilidades relacionadas a la política educacional, liderazgo y práctica.

De entre las tendencias de impacto de medio plazo, figura transformación del alumno consumidor de contenido en alumno productor de contenido. Se trata de una propuesta que tiene como objetivo integrar el proceso de producción de materiales en sala de clase, como tutoriales, vídeos y juegos digitales, para formación de una red de compartimento de contenido. Esa práctica permite abordar cuestiones relacionadas a la producción intelectual del alumno, pues al idealizar un material nuevo, el discente pasa a tener responsabilidad y derechos sobre su producción.

Este contexto abre posibilidades para abordaje de una práctica que muchos especialistas han defendido como significativa para desarrollo de la creatividad y optimización en la solución de problemas: el uso de lenguajes de programación y el pensamiento computacional en la educación básica. Aunque haya educadores escépticos con relación a los beneficios que la adopción de lenguajes de programación puede promover en el proceso educativo, hay experiencias en países

de primer mundo, como Noruega y Estados Unidos, que evidencian los beneficios que esa práctica puede ocasionar.

Wing (2006) afirma que el pensamiento computacional es una habilidad necesaria a todos los seres humanos, no solo a científicos del área de computación, bajo la justificación de que en la resolución de problemas pueden ser utilizados diversos conceptos de la computación. Según la autora, en el pensamiento computacional se usa la “elección de una representación adecuada para un problema o modelar aspectos relevantes de un problema para tornarlo tratable”. (Wing, 2006, p. 33), es decir, la autora sugiere que “pensar computacionalmente” es una necesidad humana para solucionar problemas en distintos ámbitos.

Filosofía semejante es presentada en la propuesta curricular “CSTA K–12 *Computer Science Standards*”, donde se sugiere no solo utilizarse la programación o el pensamiento computacional, sino también integrar otras habilidades, conforme ilustrado en la Figura 2.

**Figura 2:** Habilidades a desarrollar en computación



Fuente: CSTA, 2011.

En especial, con relación a las matemáticas, se tiene la oportunidad de evidenciar algo que comúnmente es cuestionado por los alumnos: la finalidad y aplicabilidad de determinados temas abordados en el salón de clase. Al utilizarse lenguajes de programación para modelar y simular soluciones de problemas, se evidencia la necesidad de utilización de simbología algébrica para variables e incógnitas.

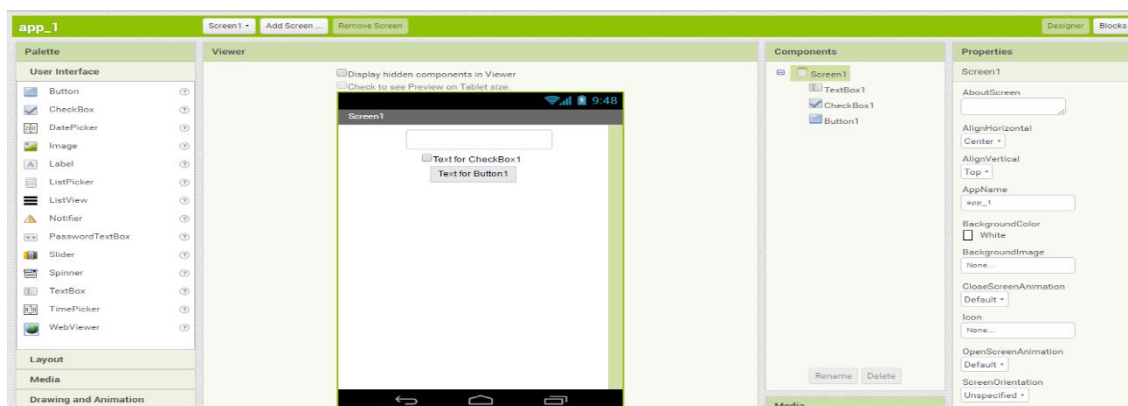
Por tratarse de una práctica que raramente está presente en la formación de profesores, utilizar lenguajes de programación para explotación de elementos y relaciones matemáticas puede convertirse en un desafío aparentemente difícil de superar, con el poco tiempo disponible para que el docente se capacite y pueda poner ese nuevo conocimiento en práctica en sala de clase. Sin embargo, los primeros pasos para implementar el pensamiento computacional pueden ser concretizados por medio de herramientas alternativas, que no necesariamente exijan conocimientos complejos sobre arquitectura de *softwares*, como por ejemplo, el *Scratch* y el *App Inventor*, ambos bajo tutela de profesionales del MIT (*Massachusetts Institute of Technology*).

### Sobre el *App Inventor* y posibilidades para la educación básica

El *App Inventor* es una herramienta de desarrollo de aplicaciones para *Android* idealizada con el propósito de democratizar y popularizar el desarrollo de este tipo de *software* entre el público joven, diferenciándose de las demás herramientas de desarrollo de aplicaciones por presentar una interfaz que facilita el entendimiento de estructuras lógicas y por la inexigibilidad de conocimientos previos sobre programación.

La estructuración de aplicaciones es efectuada en dos ambientes distintos. En el ambiente de diseñador es posible desarrollar el *layout* de la aplicación. La elección de elementos que compondrán la aplicación es efectuada por el método *drag-and-droop*, facilitando la organización. En la Figura 3 se ilustra este ambiente de creación.

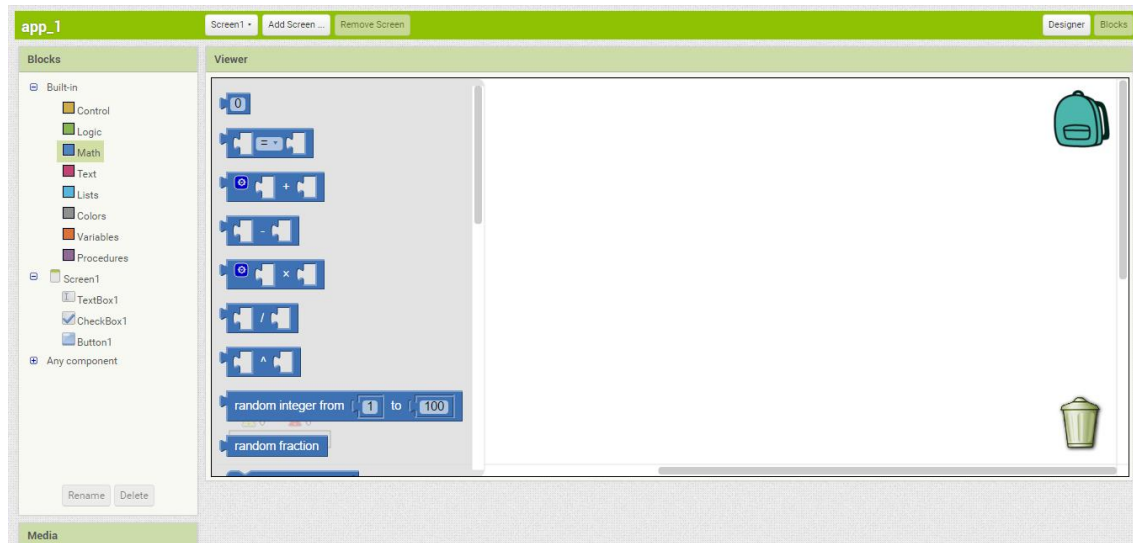
Figura 3: Ambiente *Designer* en el *App Inventor*



Fuente: Autores

En el ambiente *Blocks* se editan los comandos para el aplicativo, por medio de la combinación de bloques lógicos con encaje similar a las piezas de un rompe-cabezas (Wolber, 2011). Estos bloques son agrupados por colores, como se muestra en la Figura 4.

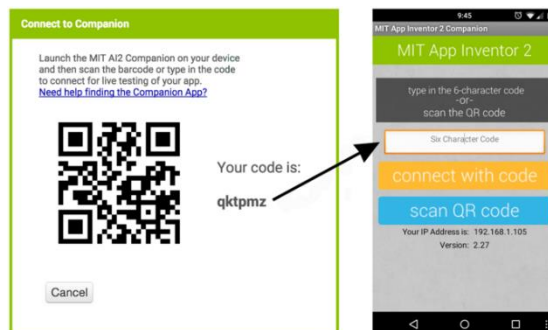
**Figura 4:** Ambiente *Blocks* en el *App Inventor*



Fuente: Autores

El proceso de desarrollo de un aplicativo puede ser visualizado en tiempo real al utilizarse una *tablet* o *smartphone* con conexión en la misma red donde el *notebook* esté conectado. Esa acción es posible por el uso concomitante de la aplicación *AI Companion*. Para que eso ocurra, basta que sea efectuada la lectura de un *QR Code*, proceso ilustrado en la Figura 5.

**Figura 5:** Proceso de lectura de *QRCode* para conexión con el *AI Companion*



Fuente: <http://appinventor.mit.edu/explore/ai2/setup-device-wifi.html>

Este proceso posibilita que el usuario controle las alteraciones y efectúe pruebas de algoritmos aún durante el desarrollo de la versión alfa de una aplicación, sin la necesidad de instalarla en un dispositivo móvil antes del término de su estructuración.

El acto de utilizarse una herramienta de creación de aplicaciones reúne características comunes tanto del pensamiento computacional citado por Wing, como del construccionismo, de Papert. Esa convergencia se origina en el hecho de que sea posibilitada la experimentación concomitantemente con la organización lógica y sólida de conocimientos sobre un determinado objeto de estudio.

En Brasil, es común que propietarios de vehículos *flex* utilicen una regla informal ampliamente difundida para comparar el costo-beneficio de los combustibles etanol y gasolina: se divide el valor del litro de etanol por el valor del litro de la gasolina. Si el resultado es menor de lo que 0,7, es ventajoso abastecer con etanol. Caso contrario, es ventajoso abastecer con gasolina. Esa regla es basada en el hecho de que el rendimiento medio del etanol es cerca del 70% del rendimiento de la gasolina.

Aunque haya un ejemplo trivial, además de la síntesis y organización lógica de informaciones relevantes sobre una situación-problema, se torna necesaria la utilización de descripción algébrica de las posibilidades. Esta característica permite contextualizar el uso de distintas simbologías para modelar un problema de forma genérica.

Se puede averiguar esa organización lógica de informaciones al analizarse la producción discente decurrente del proceso de desarrollo de una aplicación referente a esta temática. Se ejemplifica eso en la Figura 6, donde se ilustra la organización lógica de datos y la posterior implementación en el formato de comandos para ejecución de cálculos en una aplicación vía *App Inventor*, elaborada por un alumno de enseñanza media.

**Figura 6:** Ejemplo de transcripción de informaciones para la programación visual



$$\frac{A}{G} > 0,7 \rightarrow \text{Gasolina}$$

$$\frac{A}{G} < 0,7 \rightarrow \text{Alcohol}$$

$$\frac{A}{G} = 0,7 \rightarrow \text{Equivalentes}$$



```

when Button2 Click
do
  if [TextBox1.Text = ]
  then set TextBox1.Text to 0
  if [TextBox2.Text = ]
  then set TextBox2.Text to 0
  if [TextBox2.Text * 0.7 < TextBox1.Text]
  then set Label2.Text to "Ventajoso: GASOLINA"
  else if [TextBox2.Text * 0.7 = TextBox1.Text]
  then set Label2.Text to "Ventajoso: INDIFERENTE"
  else set Label2.Text to "Ventajoso: ETANOL"
  
```

Fuente: Autores

Se observa que el modelo computacional idealizado por el alumno no contempla apenas las posibilidades inicialmente enumeradas, pero trata también de posibles problemas que eventualmente podrían ocurrir, como la ausencia de datos en “TextBox1” o en “TextBox2”, campos de entrada de datos de los valores del litro de los combustibles deberían ser informados. Otra característica interesante es la transcripción de esas informaciones con formato distinto del inicialmente descrito. Se nota que la comparación entre los valores de los litros de los combustibles fue efectuada por la multiplicación del factor 0,7 por el valor del litro de la gasolina, y no por el cálculo de la razón entre los valores de los litros de los combustibles.

Sin embargo, para que ese tipo de actividad pueda ser desarrollado con éxito, contribuyendo efectivamente para el proceso de atribución de significado para las matemáticas y construcción de conocimiento, es necesario un cambio comportamental tanto del docente como del educando bajo su supervisión. Es relevante que el educando comprenda que el proceso de análisis de una situación-problema no se constituye en la repetición de pasos que serán transmitidos por el profesor. En contrapartida, el profesor debe estar preparado para tratar adecuadamente los eventuales imprevistos que puedan surgir durante el desarrollo de este tipo de actividad.

Se torna necesario un cambio de comportamiento, de forma que el docente adentre en lo que Borba y Penteadó (2012) llaman de “zona de riesgo”, donde el imprevisto impera. Concomitantemente, debe ser estimulada la postura activa y crítica del alumno, de forma que el diálogo entre pares y la argumentación racional sean integrados para presentación de respuestas.

Para eso, se sugiere organizar actividades basadas en el ciclo investigación-ideación-prototipo, proceso colaborativo donde:

- a) Los contenidos necesarios para solucionar un problema son alentados por el profesor, que orientará a los alumnos mediante estructuración de un guion de estudios sobre uno o más temas específicos;
- b) El trabajo discente pueda ser organizado en pares, para compartir saberes y discusión de estrategias de solución;
- c) Las soluciones presentadas posibiliten explicitar y compartir diferentes formas de organizar el pensamiento.

Por tratarse de un proceso de creación libre, se espera que el alumno no solo estructure una aplicación, pero efectuara pruebas para averiguación de eventuales errores en el proceso de su estructuración. Errores ocurridos durante este proceso no deben desanimar al alumno, pueden ser, incluso, utilizados como motivadores para el aprendizaje. Es importante que tanto el profesor como el alumno comprendan esos errores como parte significativa del proceso de aprendizaje, de forma que el error no sea encarado como algo negativo, más bien como generador de nuevas discusiones y reflexión sobre el objeto de estudio.

Note que la propuesta de desarrollo de aplicaciones en el salón de clase permite no solo la explotación de temas presentes en el currículum de la educación básica, también contempla parte de las habilidades expuestas como factibles en el horizonte educacional para los próximos años, principalmente en lo que se refiere a la transición del alumno de consumidor a productor.

### **Consideraciones Finales**

Se refuerza que estudios sobre implicaciones del uso de lenguajes de programación en la construcción de conocimiento matemático no es algo reciente, una vez que Papert desarrolló las primeras experiencias con este lenguaje. Luego aún en la década de 50, obteniendo éxito en una época en que la tecnología no era significativamente presente en la sociedad como ocurre en la actualidad.

Conforme a lo indicado en el NMC *Horizon Report*, el panorama educacional del siglo XXI permite que el docente pueda explotar distintas herramientas con el fin de desarrollar la investigación colaborativa en el salón de clase, con objetivo a desarrollar habilidades relacionadas a la autonomía y aprendizaje profundo.

Sin embargo, se vigorece que, aunque la adopción del uso de tecnologías en el proceso educativo formal integre múltiples aspectos de distintas vertientes educacionales y contemple distintos presupuestos teóricos, se debe tener en cuenta que solo utilizar tecnologías en sala de clase no es un factor determinante para la promoción del aprendizaje significativo. Se torna necesario que el profesor elija las herramientas tecnológicas que pretende utilizar con base en objetivos a desarrollar, conforme a lo indicado por Giraldo; Caetano y Mattos (2013).

Otro factor a ser llevado en consideración es la necesidad de la constante actualización docente, que la escasez de políticas públicas para la formación de profesores no limite las necesidades reales del salón de clase. La velocidad con que las tecnologías evolucionan exige un nuevo perfil de profesionales de la educación, de forma que estos puedan acompañar el ritmo con que la generación discente actual busca, procesa e interpreta informaciones.

## Referencias bibliográficas

- Borba, M. de C.; Pentead, M.G. (2012). *Informática e Educação Matemática*. 5. ed. Belo Horizonte: Autêntica, (Colección Tendências em Educação Matemática, 2).
- Campos, F. R. (2013). Paulo Freire e Seymour Papert: educação, tecnologias e análise do discurso. Curitiba: CRV.
- Cox, K. K. (2003). *Informática na educação escolar*. Campinas: Autores Associados. (Coleção Polêmicas do nosso tempo, 87).
- Giraldo, V; Caetano, P; Mattos, F. (2012). *Recursos Computacionais no Ensino de Matemática*. Rio de Janeiro: SBM (Colección PROFMAT, 06).
- Johnson, L., Adams B. S., Estrada, V., y Freeman, A. (2015). *NMC Horizon Report: Edição Educação Básica 2015*. Austin, Texas: The New Media Consortium.
- Mendes, I.A. (2009) *Matemática e investigação em sala de aula: tecendo redes cognitivas na aprendizagem*. 2. ed. rev. y aum. São Paulo: Livraria da Física.
- Prensky, M. (2001). *Digital Natives, Digital Immigrants*. On the Horizon, v. 9, n. 5, out.

Wing, J. M. (2006). Computational Thinking. *Communications of the ACM*, v. 49, n.3, mar.

Wolber, D. et al. (2011). *App Inventor: Create your own apps*. Sebastopol: O'Reilly.