

Objetivo

Desarrollar una aplicación que permita crear patrones geométricos como una forma de apoyo a la enseñanza de la programación.

Introducción

En los últimos años se ha incrementado en México y alrededor del mundo el impulso a la enseñanza de la Ciencia, la Tecnología, la Ingeniería y las Matemáticas (*STEM* por sus siglas en inglés), con el objetivo de que las personas desarrollen una serie de habilidades relacionadas con el uso y desarrollo de tecnologías. En México existen asociaciones enfocadas específicamente a las ciencias computacionales, con el objetivo de enseñarle a niños, niñas y jóvenes a programar, pero las herramientas con las que se cuenta son pocas en el idioma español ya que aproximadamente el 80% de las aplicaciones están en inglés. De acuerdo con *The STEM Center* de la Universidad de *Texas Austin College of Education* [1] elegir un IDE con una curva de aprendizaje poco pronunciada ayuda a los estudiantes a utilizarlo, agiliza el proceso de aprendizaje y les brinda una mayor confianza. Por todo lo anterior, en este trabajo se aborda el diseño de un IDE que permita a niños, niñas y adolescentes aprender conceptos sobre programación y matemáticas mientras crean patrones geométricos, los cuales podrán visualizar al ejecutar el programa.

Arquitectura

En la Figura 1 se presenta la arquitectura general del IDE, en gris se indicaron los componentes más relacionados con el comportamiento del IDE y en blanco los componentes relacionados con la compilación del programa resultante en el lenguaje de diseño gráfico a crear.

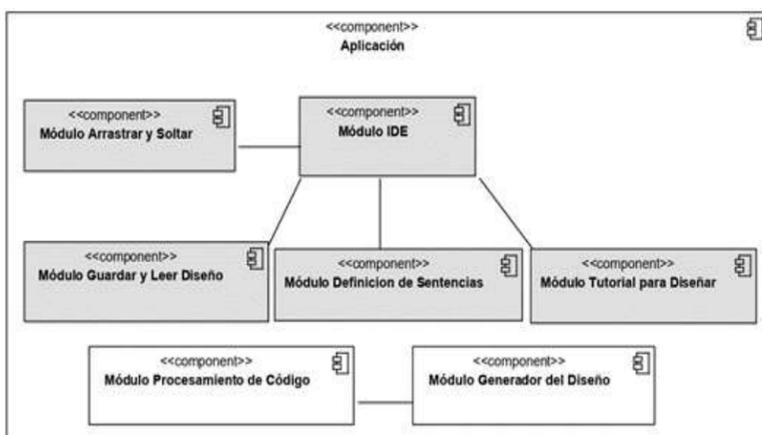


Figura 1. Componentes de la arquitectura para el ambiente de desarrollo integrado

- **Definición de Sentencias.** En este componente se definirán las sentencias posibles a utilizar por el usuario para generar algún diseño, utilizando palabras clave y palabras reservadas. haciendo uso de sintaxis propia.
- **Arrastrar y Soltar.** Es el encargado de brindar funcionalidad de *drag and drop* dentro del editor, lo que permite: arrastrar, soltar y posicionar la estructuración de código.
- **Tutorial para Diseñar.** Este componente sirve para explicar ejemplos con instrucciones a seguir por el usuario para que éste entienda la funcionalidad de la aplicación al mismo tiempo en el que crea diseños sencillos, para que posteriormente tenga la facilidad de generar sus propios diseños.
- **Guardar y Leer Diseño.** El componente “Guardar” se utiliza para conservar los cambios que se realizaron en el diseño mientras se trabajaba en él, para así seguir utilizándolo posteriormente cuando se solicite gracias al componente “Leer Diseño”.
- **Procesamiento de Código.** Es el encargado de traducir un programa de computadora escrito en un idioma particular en un programa equivalente en otro idioma (lenguaje de máquina). La traducción del programa siempre se realiza en su totalidad antes la ejecución.
- **Generador del Diseño.** Es el encargado de recibir las instrucciones procesadas o traducidas para generar el diseño y presentarlo visualmente.

Conclusiones.

Al realizar la investigación para el presente trabajo, se identificaron varios puntos relevantes a) una aproximación temprana a la programación es una ventaja competitiva, b) los ambientes gráficos resultan más atractivos para los jóvenes y se despierta mayor interés cuando el resultado es algo visible, c) existe una amplia tradición de bordados en México, d) el uso de patrones geométricos fortalece el pensamiento lógico matemático y e) en México el porcentaje de estudiantes con conocimientos suficientes del idioma inglés es reducido.

Referencias

- [1] “Ask John: Which IDE should I use with my students?”. <https://sites.utexas.edu/weteachcs/ask-john-which-ide-should-i-use-with-my-students/> (consultado jun. 24, 2020).

Diseño del IDE

Como segundo punto del diseño del IDE se realizó el maquetado de la aplicación, considerando tanto la facilidad de uso como los requerimientos no funcionales identificados. En la Fig. 2 se muestra el maquetado obtenido.

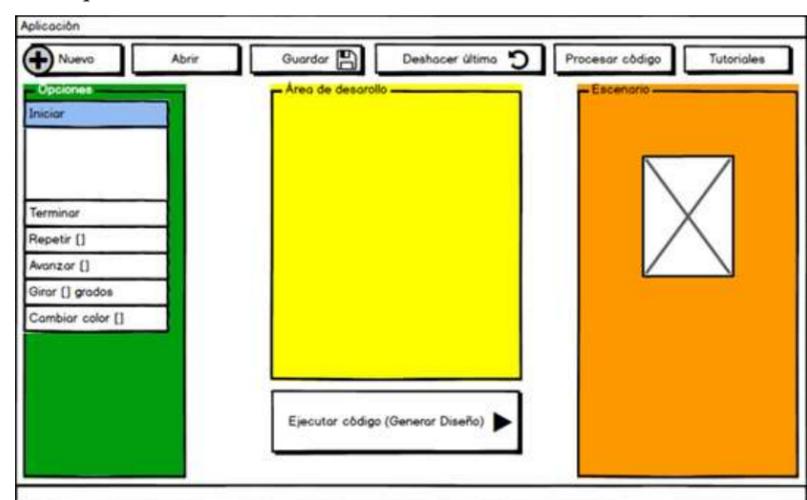


Figura 2. Maquetado del ambiente de desarrollo integrado para la enseñanza de la programación

Resultados

Al seguir los lineamientos de la Ingeniería de Software fue posible construir un ambiente de desarrollo integrado gráfico, que considera los aspectos necesarios para incorporar exitosamente a más niños, niñas y adolescentes a la tecnología disminuyendo las brechas por idioma que se presentan con otras herramientas similares. En la Figura 3, se muestra una prueba con la herramienta en su fase final.

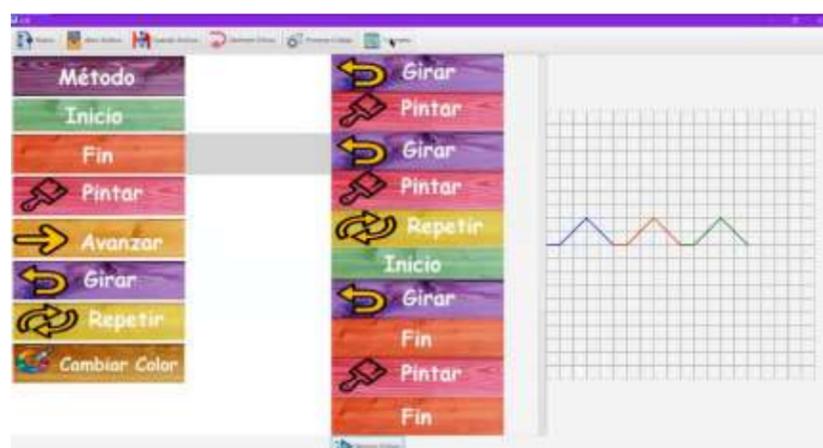


Figura 3. Prueba de la herramienta desarrollada