

Más allá de la funcionalidad

Natalia Orozco Delgado

Daniel Méndez Romero

Evelyn Vanessa Ulate Obando

Así como los artistas, los programadores pasan por un proceso creativo y de inspiración directamente vinculado con su objetivo. Si bien tradicionalmente se separan por completo los conceptos de tecnología y arte, están estrechamente relacionados y, al tratarlos como ajenos, se minimizan los beneficios de establecer tal vínculo. Tratar la programación como un arte en sí misma brinda una perspectiva totalmente distinta a la apreciación y elaboración de código; no se trata de un proceso mecánico con parches para solucionar cualquier problema, por el contrario, es elegante y propio de artistas.

Como afirma Didier Verna (2018), el arte de la programación es algo bello en sí mismo, se comprende el valor en la elegancia de una solución y el cómo es digno de ser estudiado. De esta manera, no se enfoca únicamente en el aspecto funcional de un programa, sino que también integra un componente estético, de claridad, orden y mantenibilidad. Se podría pensar que, bajo estos ideales, los profesionales siempre buscan soluciones óptimas y elegantes. Sin embargo, esta visión artística de la programación suele entrar en conflicto con la realidad de la industria tecnológica contemporánea, donde predomina la presión por entregar resultados rápidamente, “apagar incendios” y priorizar la producción constante sobre la calidad del proceso creativo.

Un código es estético no solo por su aspecto visual, sino también por su coherencia, facilidad de comprensión y modularidad. Esto engloba la organización interna del proyecto, su documentación y elementos que mantengan su sencillez y funcionalidad. Cuando un programa está bien estructurado y explicado, se favorece la colaboración entre profesionales alrededor del mundo. De igual manera, tales prácticas favorecen el sentido de comunidad en el área y contribuyen a la elaboración de material didáctico de utilidad para principiantes.

Precisamente porque el código puede poseer valor estético y comunicativo, resulta preocupante la creciente normalización de prácticas que priorizan únicamente la

funcionalidad inmediata. El problema con la ideología del desecho y practicidad no se limita al trasfondo ético en aspectos laborales o académicos, sino que promulga una actitud de indiferencia hacia las creaciones actuales y futuras, no se busca crecer ni aprender de proyectos anteriores, inspirar o comunicar un mensaje. Sin embargo, no es culpa exclusiva de las personas programadoras o profesionales en *software*, sino que, la sociedad actual promulga una ideología que deposita el valor de un producto exclusivamente en su capacidad de producir capital. Por lo tanto, se evalúa arduamente la rapidez de elaboración y puesta en producción, reducir tiempo y recursos, bajo el único principio de generar tanto dinero como sea posible. Al calificar el trabajo bajo esos parámetros, se termina devaluando el trabajo hecho con dedicación y considerable inversión humana, limitando el potencial de los desarrolladores, reduciendo su oficio a una tarea mecánica y repetitiva, que concluye con la exclusión de su origen artístico.

Esta reducción del *software* a un producto meramente funcional no solo afecta la calidad técnica del código, sino también su capacidad de transmitir identidad y significado cultural. Tradicionalmente, se suele ver a la tecnología como un causante de la pérdida de cultura en nuevas generaciones. Según Rodríguez-Cadena (2019), las tecnologías digitales han propiciado el aislamiento de las personas, causando un desarraigo social y cultural. Sin embargo, esta superficialidad que se asigna a la tecnología es un atributo asignado, y no corresponde necesariamente al carácter implícito de esta misma.

El desarraigo cultural que expresa Rodríguez-Cadena responde ante el fenómeno del desarrollo tecnológico acelerado, que no ha permitido una adaptación apropiada a las transformaciones que acarrea, tornando difícil visualizar que la tecnología trae consigo cambios culturales que deben ser abordados desde una perspectiva diferente, y que no necesariamente el cambio implica pérdida. Siendo así, resulta posible dar un giro a las tendencias, transformando la manera en la que se emplean estas nuevas herramientas, para propiciar un uso que promueva la integración de las personas con los espacios culturales y de reflexión.

La relación entre tecnología y cultura puede observarse claramente en el desarrollo de videojuegos con identidad local, donde el *software* deja de ser

únicamente entretenimiento y se convierte en una herramienta de representación cultural. Un ejemplo claro de cómo un videojuego puede convertirse en obra cultural, es el caso de *Don Memo*, un videojuego costarricense en desarrollo, concebido como un *pixel platformer 2D*. La historia sigue a un valiente toro que recorre Costa Rica para rescatar a su nieta y enfrentar al *Pisuicas*, mientras combate criaturas inspiradas en leyendas y elementos propios del imaginario nacional. Más allá de ser un producto de entretenimiento, el juego se construye como una reinterpretación digital del folclore costarricense, trasladando mitos, símbolos y referencias locales al lenguaje interactivo del *software*.

La decisión de desarrollar el juego en formato 2D con estética *pixel art* no se debe a una limitación técnica o tecnológica, sino a una elección segura sobre una expresión artística. El pixel art surge desde los orígenes de los videojuegos, trayendo una memoria colectiva digital, que al mismo tiempo funciona como lienzo para reinterpretar elementos culturales propios. En este caso, la tecnología no sustituye la cultura, sino que la traduce, y reinterpreta. Cada *sprite* diseñado, cada escenario ambientado en paisajes nacionales y cada enemigo inspirado en leyendas populares implican un proceso creativo que deja atrás la simple programación, y da pie a la observación del entorno.

Por consiguiente, el componente artístico visible y la narrativa del videojuego depende completamente de una estructura técnica subyacente: el código. Aunque invisible para el jugador, este constituye el medio que permite materializar la experiencia cultural y estética propuesta por los desarrolladores, pues es el código el que define la física de los saltos, las colisiones, las capacidades de los enemigos, y de qué manera va fluyendo la historia.

Es aquí donde se puede retomar la pregunta central: si consideramos proyectos como *Don Memo* únicamente como productos de consumo digital, su valor termina cuando deja de ser lucrativo. Pero si los entendemos como obras culturales, entonces su preservación, incluido su código fuente, se vuelve relevante, aunque el público sea de nicho. No se trata solo de conservar un videojuego, sino de cuidar una manifestación del pensamiento creativo nacional en formato digital. En esa transición de producto a patrimonio es donde cambia nuestra relación con la tecnología, dejamos

de consumir pasivamente y comenzamos a reconocerla como parte de nuestra identidad cultural.

En este punto, es importante vincular el desarrollo de videojuegos como *Don Memo* con el concepto de preservación digital del patrimonio cultural inmaterial. Isa et al. (2018) sostienen que las tecnologías digitales pueden funcionar como herramientas clave para documentar, transmitir y revitalizar expresiones culturales que, de otro modo, podrían perderse con el paso del tiempo. Si bien tradicionalmente la preservación cultural se asociaba con archivos físicos o registros escritos, hoy los entornos interactivos permiten algo más profundo, experimentar la cultura. Un videojuego ambientado en Costa Rica, que incorpora leyendas como el *Pisuicas*, gastronomía típica y símbolos identitarios, no sólo representa cultura, sino que la convierte en una experiencia participativa. En este sentido, el código deja de ser únicamente una estructura funcional y se transforma en un medio de conservación cultural; cada línea programada sostiene una narrativa, una memoria colectiva y una reinterpretación contemporánea del folclore nacional. Así, el acto de programar adquiere una dimensión patrimonial y artística.

Por otro lado, cabe destacar que buscar un enfoque más cultural o artístico no implica sacrificar la eficiencia o funcionalidad del código, y tampoco busca promover la supremacía de la estética. Proponer una visión artística de la tecnología pretende que el enfoque sea primordialmente el proceso de creación, más que el resultado esperado; significa que el código debe hacerse y rehacerse las veces que sea necesario, aunque en términos de funcionalidad cumpla con las expectativas, porque el arte surge de esos pequeños detalles que, quizá no hacían falta, pero son esas sutilezas que generan una conexión auténtica con el observador, y dotan de identidad propia al artista.

Asimismo, apostar por el arte significa dejar de lado la competitividad. Hoy en día, se generan una infinidad de productos, tecnologías y código con el único propósito de ser mejor que otro. Muchas veces, incluso se margina la originalidad, para solo copiar la fórmula de éxito que a alguien más se le ocurrió, y de esa manera tener una parte del pastel para sí. Desarrollar cualquier cosa que tenga como fin primario superar a los demás para beneficio propio, se despoja por sí mismo del valor desinteresado del

arte y la cultura. Y, desafortunadamente, esta es la máxima que rige en el mundo moderno.

Asimismo, comprender la programación como una práctica artística también implica reconocer su dimensión colectiva. A diferencia de la visión individualista predominante en la industria tecnológica, movimientos como el código abierto fomentan espacios de colaboración donde el conocimiento, las ideas y las soluciones pueden compartirse y perfeccionarse de manera comunitaria, permitiendo una buena integración social.

En cuanto a esta integración social, estos espacios permiten formar una comunidad invaluable para muchas personas, permitiendo adquirir nuevas experiencias, pues estas comunidades logran atraer tanto a personas experimentadas, como a principiantes, de diferentes países e idiosincrasias. Justamente, el código abierto permite que este tipo de interacciones contengan también un carácter de formativo a nivel técnico y reflexivo, pues se suele compartir material para aprendizaje, iniciar foros de discusión y consultar directamente a personas con mayor experiencia, promoviendo una mezcla enriquecedora de perspectivas, ideas, soluciones y aprendizaje.

Sin embargo, este ideal colaborativo suele verse limitado por dinámicas laborales que priorizan productividad constante y competencia acelerada. En muchos casos, las largas jornadas y la presión empresarial terminan sofocando cualquier posibilidad de experimentación creativa o desarrollo artístico dentro del *software*. Es por este principio que se obliga a los trabajadores a trabajar largas jornadas, que acaban ahogando cualquier pizca de entusiasmo o creatividad, y que culmina reduciéndolos a simples números u objetos desechables. Y es entonces cuando se deja de vivir, y ahora solo se puede sobrevivir. Sin embargo, ¿cómo se podría hacer arte, si ya no se está vivo? Por ende, un enfoque artístico implicaría sacrificar el avance acelerado, y moverse con paciencia.

Así, replantear la programación como una forma de arte no significa abandonar la funcionalidad o la eficiencia, sino reconocer que el *software* también puede contener valor cultural, humano y creativo. Esta perspectiva permite comprender el código no solo como una herramienta de producción, sino como una manifestación contemporánea de identidad y memoria colectiva.

Particularmente, resulta importante comprender que la tecnología no necesariamente aleja al individuo de la cultura, como suele verse desde ciertas perspectivas críticas. Por el contrario, cuando el desarrollo tecnológico incorpora intención creativa y sensibilidad humana, puede convertirse en un medio para preservar identidades, transmitir tradiciones y fomentar nuevas formas de expresión cultural. Desde esta visión, el código deja de entenderse únicamente como una herramienta técnica y pasa a concebirse también como un lenguaje con propósito, capaz de comunicar ideas, valores y experiencias.

En consecuencia, reconocer la programación como una forma de arte implica aceptar que detrás de cada sistema existe una intención que trasciende la simple ejecución de tareas. La funcionalidad y la eficiencia continúan siendo importantes, pero pueden coexistir con la creatividad, la estética y el componente cultural. Solo así es posible construir una relación más humana con la tecnología, donde el desarrollo de *software* no represente únicamente producción, sino también expresión y memoria colectiva.

Referencias

Isa, W., Zin, N., Rosdi, F., & Sarim, H. (2018). Digital Preservation of Intangible Cultural Heritage. *Indonesian Journal of Electrical Engineering and Computer Science*.
<https://doi.org/10.11591/ijeecs.v12.i3.pp1373-1379>.

Rodriguez-Cadena, R. (2019). Tecnología digital y afectaciones a la cultura de aprendizaje de sujeto social. *Revista Venezolana de Gerencia*, 2, 502-514.
<https://www.redalyc.org/journal/290/29063446027/29063446027.pdf>

Verna, D. (2018). Lisp, Jazz, Aikido--Three Expressions of a Single Essence. *The Art, Science, and Engineering of Programming*, 2 (3), 10-50.
<https://doi.org/10.22152/programming-journal.org/2018/2/10>