PYTHON LLEGA FUERTE A LAS APLICACIONES MÓVILES

Cuando se pensaba en Python, se veía como un lenguaje de backend, de ciencia de datos y de Inteligencia Artificial. Esto se puede acabar con Kivy y BeeWare, que lideran la investigación para llevar las aplicaciones al móvil, tanto Android como iOS.

Kivy, el pionero en la interfaz táctil

Lanzado originalmente en 2011, Kivy se consolidó como el primer framework que permitía a los desarrolladores de Python crear aplicaciones realmente multiplataforma, siendo multitáctil y con un interfaz multimedia. Es un proyecto de código abierto y capaz de operar en una amplia gama de sistemas operativos.

Su principal característica es que Kivy no utiliza los widgets nativos del sistema operativo, en su lugar, utiliza su propio motor de renderizado basado en OpenGLES (Graphics Library for Embedded Systems). La principal ventaja de este enfoque es que garantía una consistencia visual en todas las plataformas. Lo que se ve en Windows, se ve en Android, iOS o Linux.

Para la construcción de la interfaz de usuario, Kivy utiliza el KV Language. Este lenguaje de marcado declarativo (como HTML) permite a los desarrolladores separar la lógica de programación (escrita en Python) del diseño de la interfaz (escrito en .kv), resultando un código limpio, modular y fácil de mantener.

La mayor fortaleza de Kivy está en su flexibilidad para el desarrollo de aplicaciones que requieran una manipulación táctil avanzada o un diseño muy personalizado.

Su dependencia de OpenGL lo convierte en una buena opción para pequeños juegos 2D donde la velocidad de renderizado es importante. Es bastante utilizado para construir interfaces de usuario interactivos o soluciones de punto de venta, donde el control multitáctil es importante.

Otra de sus características es que no es muy complicado su aprendizaje en la parte de diseño, permitiendo crear y desplegar rápidamente prototipos funcionales en distintas plataformas.

BeeWare, la apuesta por la experiencia nativa

BeeWare o es un único framework, sino una colección de herramientas que vieron la luz en 2013. Su vocación es clara y ambiciosa, permitir a los desarrolladores de Python construir aplicaciones nativas para cada plataforma utilizando un solo código.

El principio fundamental de BeeWare es que la aplicación debe verse, sentirse y comportarse como una aplicación nativa. Para lograr esto, se apoya en dos componentes principales:

Toga: Esta es la librería de widgets de BeeWare. A diferencia de Kivy, Toga traduce el código de Python a llamadas a los widgets nativos del sistema operativo. Por ejemplo, un botón de Python se cambia como un UIButton en iOS o un antroid.widget.Button en Android, o un botón estándar de Windows.

Bridgeware (Wrappers): Utiliza una serie de envoltorios (wrappers) y compiladores como Briefcase, para empaquetar el código de Python de forma que se pueda ejecutar como una aplicación nativa en la plataforma adecuada.

Lo que ofrece esta plataforma es dar a Python la capacidad de poder instalarse en cualquier sistema operativo. Su proyección de futuro es enorme, ya que resuelve el principal problema que tiene Python en el desarrollo de aplicaciones a usuarios, la falta de experiencia nativa.

Las aplicaciones creadas con BeeWare son familiares para los usuarios de cada plataforma, ya que utilizan los elementos de interfaz a los que están acostumbrados (fuentes, estilo, componentes, comportamientos, etc.)

Además, es capaz de integrarse en bastantes sistemas operativos, como Windows, macOS, Linux, iOS, Android y también es capaz de crear una web a través de WebAssembly. Esto permite a las empresas y desarrolladores apuntar

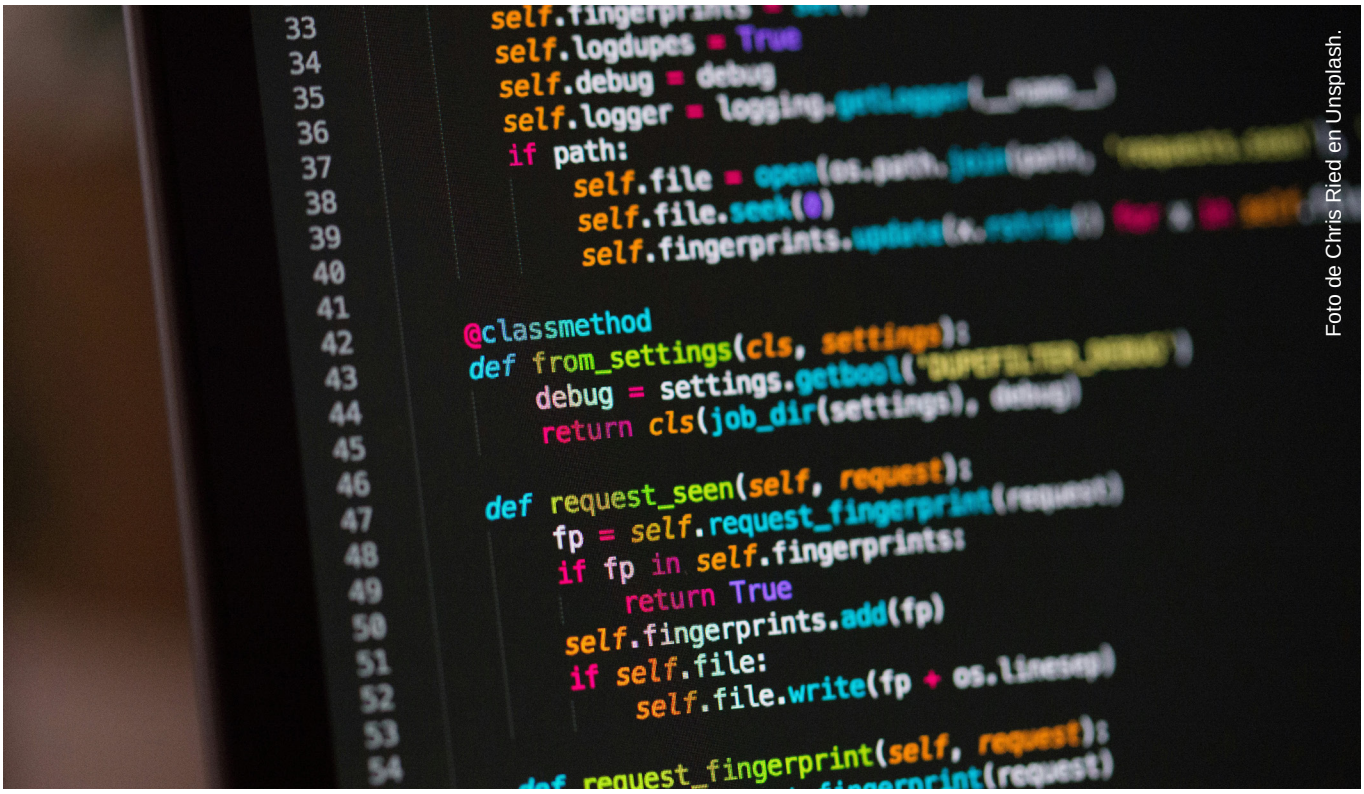

Foto de Chris Ried en Unsplash.

> BeeWare y Kivy están borrando las líneas del desarrollo móvil y de escritorio, permitiendo a los desarrolladores de Python construir aplicaciones multiplataforma con un solo código.

a los principales sistemas del mercado con un solo código fuente.

Otra de las ventajas de BeeWare es que usa las características mas modernas de Python, evitando dependencias heredadas, lo que le permite mantenerse ágil ante cualquier evolución.

Esta plataforma, todavía se debe considerar muy joven, pero su proyección hacia la experiencia nativa lo posiciona como el futuro del full-stack en Python.

Tanto Kivy como BeeWare suponen un hito importante en la evolución de Python. Ambos han roto la barrera de la ejecución en sistemas operativos dispares y, de paso, has desafiado la noción de que el desarrollo móvil y desktop deben estar dominado por lenguajes nativos o frameworks basados en JavaScript.

Gracias a herramientas como estas, un desarrollador que ya domina Python para machine learning, automatización o backend, puede ahora extender sus desarrollos para crear aplicaciones visuales para usuario final con una mínima curva de aprendizaje adicional. El objetivo es claro, un único código fuente en Python que sea capaz de llegar a cualquier dispositivo. ■

Javier Gómez Delgado es autor del libro *Programación en Python* publicado por ESIC Editorial.