



# El Tao de la Programación

Traducido por Geoffrey James, Transcrito por Seth Robertson, Versión Española por TESI

## Tabla de Contenido

- [Libro 1 - El Vacío Silencioso](#)
- [Libro 2 - Los Maestros Antiguos](#)
- [Libro 3 - Diseño](#)
- [Libro 4 - Codificación](#)
- [Libro 5 - Mantenimiento](#)
- [Libro 6 - Administración](#)
- [Libro 7 - Sabiduría Corporativa](#)
- [Libro 8 - Hardware y Software](#)
- [Libro 9 - Epílogo](#)

## Libro Uno: El Vacío Silencioso

Así habló el maestro Desarrollador:

"Cuando hayas aprendido a sacar el código de error, será la hora que te vayas."

### 1.1

Algo misterioso se forma, nace en el vacío silencioso. Esperando solo e inmóvil, está a la vez detenido y en movimiento constante. Es la fuente de todos los programas. Yo no sé su nombre, así que lo llamaré el Tao de la Programación.

Si el Tao es grandioso, entonces el sistema operativo es grandioso. Si el sistema operativo es grandioso, el compilador es grandioso. Si el compilador es grandioso, entonces las aplicaciones son grandiosas. El usuario está complacido y hay armonía en el mundo.

El Tao de la Programación fluye lejos y regresa con el viento de la mañana.

### 1.2

El Tao engendró al lenguaje de máquina. El lenguaje de máquina engendró al ensamblador.

El ensamblador engendró al compilador. Ahora existen diez mil lenguajes.

Cada lenguaje tiene su propósito, aunque sea humilde. Cada lenguaje expresa el Yin y el Yang del software. Cada lenguaje tiene su lugar dentro del Tao.

Pero no programes en COBOL si puedes evitarlo.

### 1.3

En el principio era el Tao. El Tao engendró al Espacio y Tiempo. Por lo tanto, Espacio y Tiempo son el Yin y el Yang de la programación.

Los Desarrolladores que no comprenden el Tao siempre se quedarán sin tiempo y espacio para sus programas. Los Desarrolladores que comprenden el Tao siempre tendrán suficiente tiempo y espacio para realizar sus objetivos.

¿Cómo podría ser de otra forma?

### 1.4

Al Desarrollador sabio le hablan del Tao y lo sigue.

Al Desarrollador promedio le hablan del Tao y lo busca.

Al Desarrollador necio le hablan del Tao y se rie.

Si no fuera por la risa, no habría Tao.

Los sonidos más altos son los más difíciles de oír.

Ir hacia el frente es una forma de retirarse.

Los grandes talentos se muestran hasta tarde en la vida.

Hasta un programa perfecto tiene errores.

## Libro Dos: Los Maestros Antiguos

Así habló el maestro Desarrollador:

"Después de tres días sin programar, la vida se vuelve sin sentido."

### 2.1

Los Desarrolladores de antes eran misteriosos y profundos. No podemos comprender sus pensamientos, por eso, todo lo que haremos, será describir su apariencia.

Despierto, como una zorra cruzando el agua. Alerta, como un general en el campo de batalla. Amable, como un anfitrión saludando a sus huéspedes. Sencillo, como bloques de madera sin tallar. Opaco, como pozos negros en cuevas oscuras.

¿Quién puede decir los secretos de sus corazones y mentes?

La respuesta solo existe en el Tao.

### 2.2

El Gran Maestro Turing una vez soñó que era una máquina. Cuando se despertó exclamó:

"¡No sé si soy Turing soñando que era una máquina, o una máquina soñando que soy Turing!"

## 2.3

Un Desarrollador de una compañía muy grande de computadoras fué a una conferencia de software y luego regresó a reportarse con su jefe, diciendo: "¿Que clase de Desarrolladores trabajan para las otras compañías? No se comportan y no les interesan las apariencias. Su pelo era largo y descuidado y sus ropas arrugadas y viejas. Arruinaron las reglas de etiqueta e hicieron ruidos groseros durante mi presentación."

El jefe dijo: "Nunca debí haberte enviado a la conferencia. Esos Desarrolladores viven más alla del mundo físico. Ellos consideran a la vida un absurdo, una coincidencia accidental. Ellos van y vienen sin conocer limitaciones. Sin ningún cuidado, ellos viven solo para sus programas. ¿Porque deberían molestar por las convenciones sociales?

Ellos viven dentro del Tao."

## 2.4

Un novicio le preguntó al Maestro: "Hay un Desarrollador que nunca diseña, documenta, ni prueba sus programas. Y todos los que lo conocen lo consideran uno de los mejores Desarrolladores del mundo. ¿Cómo es esto?"

El Maestro contestó: "Ese Desarrollador se ha convertido en un maestro del Tao. El ha ido más allá de la necesidad de diseñar; el no se enoja cuando el sistema se cae, acepta al universo sin preocupación. El ha ido más allá de la necesidad de documentar; ya no le interesa si alguien más ve su código. El ha ido más allá de la necesidad de probar; cada uno de sus programas son perfectos dentro de sí mismos, serenos y elegantes, su propósito es auto-evidente. Realmente, el ha penetrado el misterio del Tao."

## Libro Tres: Diseño

Así habló el maestro Desarrollador:

"Cuando el programa está siendo probado, es muy tarde para hacer cambios de diseño."

### 3.1

Hubo una vez un hombre que fué a una feria de computadoras. El primer día al entrar, le dijo al guardia de la puerta:

"Yo soy un gran ladrón, renombrado por mis hazañas de robar tiendas. Estás avisado de antemano, porque esta feria no escapará sin ser saqueada."

Estas palabras incomodaron mucho al guardia, porque adentro había millones de dólares en equipo computacional, así que observó al hombre cuidadosamente. Pero el hombre solamente vagaba de stand en stand, murmurando para sí.

Cuando el hombre se iba, el guardia se lo llevó aparte y buscó entre sus ropas, pero no encontró nada.

Al siguiente día de la feria, el hombre regresó y regañó al guardia diciendo: "Ayer escapé con un gran botín, pero hoy será todavía mejor." Así que el guardia lo observó más de cerca, pero sin resultados.

En el último día de la feria, el guardia no pudo resistir más su curiosidad. "Señor Ladrón, " dijo "estoy tan confundido que no puedo vivir en paz. Por favor ilumíneme. ¿Qué es lo que está robando?"

El hombre sonrió. "Estoy robando ideas" dijo.

### 3.2

Hubo una vez un maestro Desarrollador que escribía programas no estructurados. Un Desarrollador novicio, buscando imitarlo, también comenzó a escribir programas no estructurados. Cuando el novicio le pidió al maestro evaluar su progreso, el maestro lo criticó por escribir programas no estructurados, diciendo: "Lo que es apropiado para el maestro no es apropiado para el novicio. Debes entender el Tao antes de trascender la estructura."

### 3.3

Hubo una vez un Desarrollador que estaba asignado a la corte del señor de Wu. El señor le preguntó al Desarrollador: "¿Qué es más fácil de diseñar: un paquete de contabilidad o un sistema operativo?"

"Un sistema operativo", contestó el Desarrollador.

El señor lanzó una exclamación de incredulidad. "De seguro un paquete de contabilidad es trivial comparado con la complejidad de un sistema operativo", dijo.

"No tanto," dijo el Desarrollador, "al diseñar un paquete de contabilidad, el Desarrollador opera como un mediador entre personas que tienen ideas diferentes: cómo debe operar, cómo debe entregar sus reportes, y cómo debe conformarse con las leyes de impuestos. Al contrario, un sistema operativo no está limitado por las apariencias externas. Al diseñar un sistema operativo, el Desarrollador busca la armonía más simple entre máquina e ideas. Es por esto que un sistema operativo es más fácil de diseñar."

El señor de Wu asintió y sonrió. "Eso está bien y correcto, pero ¿cuál es más fácil de depurar?"

El Desarrollador no contestó.

### 3.4

Un gerente fue con el maestro Desarrollador y le mostró el documento de requerimientos para una nueva aplicación. El gerente le preguntó al maestro: "¿Cuánto tiempo tomará diseñar este sistema si le asigno cinco Desarrolladores al proyecto?"

"Tomará un año", dijo el maestro rápidamente.

"¡Pero necesito este sistema inmediatamente o antes! ¿Cuánto tiempo se tardará en diseñar este sistema si le asigno diez Desarrolladores?"

El maestro frunció el ceño. "En este caso, tomará dos años."

"¿Y si le asigno cien Desarrolladores?"

El maestro se encogió de hombros. "Entonces el diseño nunca será completado", dijo.

## **Libro Cuatro: Codificación**

Así habló el maestro Desarrollador:

"Un programa bien escrito es su propio cielo; un programa mal escrito es su propio infierno."

### **4.1**

Un programa debe ser ligero y ágil, sus subrutinas conectadas como un collar de perlas. El espíritu e intención del programa debe ser retenido ante todo. No deberá haber ni poco ni mucho, ni ciclos innecesarios ni variables sin usar, ni falta de estructura ni excesiva rigidez.

Un programa debe seguir la 'Ley del Menor Asombro'. ¿Cuál es ésta ley? Es simplemente que el programa siempre debe responder al usuario de la manera que lo asombre lo menos posible.

Un programa, no importa cuán complejo sea, debe actuar como una sola unidad. El programa debe ser dirigido por la lógica interna y no por las apariencias externas.

Si el programa falla en estos requerimientos, estará en un estado de desorden y confusión. La única forma de corregir esto es reescribir el programa.

### **4.2**

Un novicio le preguntó al maestro: "Tengo un programa que a veces corre y a veces aborta. He seguido las reglas de la programación, y aún así estoy desconcertado. ¿Cuál es la razón de esto?"

El maestro contestó: "Estás confundido porque no entiendes el Tao. Sólo un tonto espera comportamiento racional de sus semejantes humanos. ¿Qué esperas de una máquina que los humanos han construido? Las computadoras simulan determinismo; solo el Tao es perfecto.

Las reglas de programación son transitorias; solo el Tao es eterno. Por lo tanto tu debes contemplar al Tao antes de recibir iluminación."

"¿Pero cómo sabré cuando he recibido la iluminación?", preguntó el novicio.

"Tu programa correrá entonces correctamente", contestó el maestro.

### **4.3**

Un maestro estaba explicando la naturaleza del Tao a uno de sus novicios, "El Tao está encarnado en todo software -- sin importar cuán insignificante sea", dijo el maestro.

"¿Está el Tao en una calculadora portátil?", preguntó el novicio.

"Está", fué la respuesta.

"¿Está el Tao en un video juego?", continuó el novicio.

"Está hasta en un video juego", dijo el maestro.

"¿Y está el Tao en el DOS de una computadora personal?"

El maestro tosió y cambió ligeramente su posición. "La lección se acabó por hoy", dijo.

#### 4.4

El Desarrollador del Príncipe Wang estaba codificando. Sus dedos bailaban sobre el teclado. El programa compiló sin un mensaje de error, y el programa corrió como viento ligero.

"¡Excelente!", exclamó el Príncipe, "¡Tu técnica no tiene fallas!"

"¿Técnica?", dijo el Desarrollador, girándose hacia su terminal, "Lo que yo sigo es el Tao -- mas allá de toda técnica. Cuando al principio empecé a programar yo podía ver el programa completo en un bloque. Después de tres años ya nunca más vi ese bloque. En vez de eso, usé subrutinas. Pero ahora no veo nada. Todo mi ser existe en un vacío sin forma. Mi sentidos están ociosos. Mi espíritu, libre para trabajar sin un plan, sigue su propio instinto. En resumen, mi programa se escribe así mismo. Es verdad, a veces hay problemas y dificultades. Las veo venir, me freno, observo silenciosamente. Entonces cambio una sola linea de código y las dificultades se desvanecen como nubes de humo. Entonces compilo el programa. Me siento erguido y dejo que el gozo del trabajo llene mi ser. Cierro mis ojos por un momento y entonces cierro mi sesión."

El Príncipe Wang dijo, "¡Ojalá todos mis Desarrolladores fueran tan sabios!"

## Libro Cinco: Mantenimiento

Así habló el maestro Desarrollador:

"Aunque un programa sólo tenga tres líneas de largo, algún dia tendrá que ser mantenido."

#### 5.1

Una puerta bien usada no necesita aceite en sus bisagras. Una corriente que fluye veloz no se estanca. Ni el sonido ni los pensamientos pueden viajar a través de un vacío. El software se pudre si no se usa.

Esos son grandes misterios.

#### 5.2

Un gerente le preguntó a un Desarrollador cuánto tardaría en terminar el programa en el cual estaba trabajando. "Lo terminaré mañana", respondió el Desarrollador.

"Creo que no estás siendo realista," dijo el gerente, "¿En realidad, cuánto tiempo tardará?"

El Desarrollador pensó por un momento. "Tengo algunas características que desearía agregar. Esto tomará al menos dos semanas", finalmente dijo.

"Aún esto es esperar mucho," insistió el gerente, "Estaré satisfecho si tu simplemente me avisas cuando el programa esté completo."

El Desarrollador aceptó esto.

Varios años pasaron, el gerente se retiró. En el camino a su comida de despedida, descubrió al Desarrollador dormido en su terminal. Había estado programando toda la noche.

### **5.3**

Un Desarrollador novicio fue una vez asignado para codificar un paquete financiero sencillo.

El novicio trabajó furiosamente por muchos días, pero cuando su maestro revisó su programa, descubrió que contenía un editor de pantalla, un conjunto de rutinas generales gráficas, y una interfaz con inteligencia artificial, pero ni la más remota mención de nada financiero.

Cuando el maestro le preguntó acerca de esto, el novicio se indignó. "No seas tan impaciente," dijo, "ya pondré esa cosa financiera eventualmente."

### **5.4**

¿Acaso un buen granjero abandona la cosecha que ha plantado?

¿Acaso un buen maestro pasa por alto aún al más humilde estudiante?

¿Acaso un buen padre permite que uno sólo de sus hijos se muera de hambre?

¿Acaso un buen Desarrollador se rehusa a mantener su código?

## **Libro Seis: Administración**

Así habló el maestro Desarrollador:

"Sean los Desarrolladores muchos y los administradores pocos -- entonces todos serán productivos."

### **6.1**

Cuando los administradores tienen juntas infinitas, los Desarrolladores escriben juegos. Cuando los contadores hablan de ganancias trimestrales, el presupuesto de desarrollo está a punto de ser recortado. Cuando los científicos en jefe pintan el cielo azul, las nubes están a punto de formarse.

En verdad, esto no es el Tao de la Programación.

Cuando los administradores hacen compromisos, los programas de juegos son ignorados.

Cuando los contadores hacen planes a largo plazo, la armonía y el orden están a punto de ser restaurados. Cuando los científicos en jefe señalan los problemas inmediatos, los problemas serán pronto resueltos.

En verdad, esto es el Tao de la Programación.

### **6.2**

¿Por qué los Desarrolladores no son productivos? Porque pierden su tiempo en juntas.

¿Por qué los Desarrolladores se vuelven rebeldes? Porque la administración interfiere mucho.

¿Por qué los Desarrolladores renuncian uno a uno? Porque han sido explotados.

Habiendo trabajado para una pobre administración, ellos ya no valoran sus empleos.

## 6.3

Un gerente estaba a punto de ser despedido, cuando un Desarrollador que trabajaba para él inventó un nuevo programa que se hizo muy popular y se vendió bien. Como resultado, el gerente retuvo su empleo.

El gerente intentó darle al Desarrollador un bono, pero el Desarrollador se rehusó, diciendo, "Escribí el programa porque pensé que era un concepto interesante, y por eso no espero ninguna recompensa."

El gerente, al oír esto, comentó, "Este Desarrollador, aunque tiene una posición poco estimada, entiende bien el deber propio de un empleado. ¡Déjenos promoverlo al privilegiado puesto de consultor de la administración!"

Pero cuando le dijo esto, el Desarrollador una vez más se rehusó, diciendo, "Yo existo para poder programar. Si yo fuera promovido, no haría nada excepto desperdiciar el tiempo de todos. ¿Me puedo ir ya? Tengo un programa en el que estoy trabajando."

## 6.4

Un gerente fué con sus Desarrolladores y les dijo: "Con respecto a sus horas de trabajo: ustedes van a tener que venir a las nueve de la mañana e irse a las cinco de la tarde." Con esto, todos se enojaron y varios renunciaron en el momento.

Entonces el gerente dijo: "Está bien, en ese caso ustedes pueden fijar sus propias horas de trabajo, siempre y cuando terminen sus proyectos a tiempo." Los Desarrolladores, ahora satisfechos, comenzaron a llegar a mediodía y trabajar hasta la madrugada.

## Libro Siete: Sabiduría Corporativa

Así habló el maestro Desarrollador:

"Le puedes mostrar un programa a un ejecutivo de la corporación, pero no puedes hacer que entienda las computadoras."

### 7.1

Un novicio le preguntó al maestro: "En el este existe una gran estructura arbórea que los hombres llaman 'Cuarteles Corporativos'. Esta hinchado y deformé con vice-presidentes y contadores. Lanza una multitud de memos, cada uno diciendo '¡Ve Aquí!' o '¡Ve Allá!' y nadie sabe qué significa. Cada año se ponen nombres nuevos en las ramas, pero todo sin resultados. ¿Cómo puede una entidad tan no natural existir?"

El maestro replicó: "Tu percibes esta inmensa estructura y estás perturbado de que no tenga un propósito racional. ¿No tienes entretenimiento de sus giros infinitos? ¿No disfrutas el alivio de programar sin problemas bajo sus ramas protectoras? ¿Porque te molesta su inutilidad?"

### 7.2

En el este existe un tiburón que es más grande que todos los demás peces. Se transforma en un pájaro cuyas alas son como nubes llenando el cielo. Cuando el pájaro se mueve a través de la tierra, lleva un mensaje de los Cuarteles Corporativos. Este mensaje lo deja caer en medio de los Desarrolladores, como una gaviota haciendo sus marcas en la playa. Entonces el pájaro se monta en el viento, y con el cielo azul a sus espaldas, regresa a su hogar.

El Desarrollador novicio mira sorprendido al pájaro, porque no lo entiende. El Desarrollador promedio teme la llegada del pájaro, porque tiene miedo de su mensaje. El maestro Desarrollador continúa trabajando en su terminal, porque él no sabe que el pájaro ha venido y se ha ido.

### 7.3

El Mago de la Torre de Marfil llevó su último invento para que el maestro Desarrollador lo examine. El mago acarrea una gran caja negra a la oficina del maestro mientras el maestro espera en silencio.

"Esta es una estación de trabajo integrada, distribuida y de propósito general," comenzó el mago, "diseñada ergonómicamente con un sistema operativo propietario, lenguajes de sexta generación, y múltiples interfaces de usuario avanzadas. Tomó a mis asistentes varios cientos de años-hombre construirla. ¿No es sorprendente?"

El maestro alzó sus cejas ligeramente. "Por supuesto que es sorprendente", dijo.

"Los Cuarteles Corporativos han mandado," continuó el mago, "que todos usen esta estación de trabajo como la plataforma para los nuevos programas. ¿Está de acuerdo con esto?"

"Ciertamente," replicó el maestro, "¡Haré que sea transportada al centro de datos inmediatamente!" Y el mago regresó a su torre muy complacido.

Muchos días después, un novicio vagaba por la oficina del maestro Desarrollador y dijo, "No puedo encontrar el listado de mi nuevo programa. ¿No sabes donde puede estar?"

"Sí," replicó el maestro, "los listados están apilados en la plataforma del centro de datos".

### 7.4

El maestro Desarrollador se mueve de un programa a otro programa sin miedo. Ningún cambio en la administración le puede hacer daño. No será despedido, aunque el proyecto sea cancelado. ¿Por qué es esto? El está lleno con el Tao.

## Libro Ocho: Hardware y Software

Así habló el maestro Desarrollador:

"Sin viento, el pasto no se mueve. Sin software, el hardware es inútil."

### 8.1

Un novicio le preguntó al maestro: "Percibo que una compañía de computadoras es mucho más grande que todas las demás. Está tan arriba de la competencia como un gigante entre enanos. Cualquiera de sus divisiones podría abarcar un negocio entero. ¿Por qué es así?"

El maestro replicó, "¿Por qué haces preguntas tan tontas? Esa compañía es grande porque es así de grande. Si sólo hiciera hardware nadie lo compraría. Si sólo mantuviera sistemas, las personas la tratarían como a un sirviente. Pero porque combina todas esas cosas, ¡las personas piensan que es uno de los dioses! Al no buscar intentarlo, conquista sin esfuerzo."

## 8.2

Un maestro Desarrollador pasó cerca de un Desarrollador novicio un día. El maestro notó la preocupación del novicio con un juego de computadora portátil. "Perdóname," dijo, "¿puedo examinarlo?"

El novicio prestó atención y le pasó el dispositivo al maestro. "Veo que el aparato dice que tiene tres niveles de juego: Fácil, Intermedio y Difícil", dijo el maestro. "Y todavía cada aparato tiene otro nivel de juego, donde el dispositivo no busca conquistar al humano, ni ser conquistado por el humano."

"Ruego, gran maestro," imploró el novicio, "¿cómo hace uno para encontrar esa misteriosa configuración?"

El maestro tiró el dispositivo al suelo y lo aplastó bajo su pie. Y de repente el novicio fué iluminado.

## 8.3

Había una vez un Desarrollador que trabajaba con microprocesadores. "Mira que tan bien estoy aquí," le dijo a un Desarrollador de mainframes que vino a visitarlo, "tengo mi propio sistema operativo y dispositivo de almacenamiento de archivos. No tengo que compartir mis recursos con nadie. El software es consistente y fácil de usar. ¿Por qué no dejas tu trabajo actual y te unes a mí aquí?"

El Desarrollador de mainframes entonces comenzó a describir su sistema a su amigo, diciendo: "El mainframe está sentado como un antiguo adivino meditando en medio del centro de datos. Sus unidades de disco descansan una con otra como un gran océano de maquinaria. El software es multifacético como un diamante y tan enredado como una selva primitiva. Los programas, cada uno es único, se mueven a través del sistema como un río que fluye rápido. Es por eso que soy feliz donde estoy."

El Desarrollador de microcomputadoras, al oír esto, guardó silencio. Pero los dos Desarrolladores siguieron siendo amigos hasta el fin de sus días.

## 8.4

Hardware se encontró a Software en el camino a Changtse. Software dijo: "Tu eres el Yin y yo soy el Yang. Si viajamos juntos nos volveremos famosos y ganaremos enormes sumas de dinero." Y así la pareja marchó junta, pensando en conquistar al mundo.

Luego ellos conocieron a Firmware, quien estaba vestido con harapos raídos, y cojeaba apoyado en un bastón espinoso. Firmware les dijo: "El Tao reside más allá del Yin y Yang. Es silencioso y erguido como un pozo de agua. No busca fama, por lo tanto nadie sabe de su presencia. No busca fortuna, porque es completo dentro de sí mismo. Existe más allá del espacio y el tiempo."

Software y Hardware, avergonzados, regresaron a sus casas.

## Libro Nueve: Epílogo

Así habló el maestro Desarrollador:  
"Es hora de que te marches."