

Manifiesto de Software a Medida

Lo que NO es el software a medida, y excepciones

Desarrollar software parece sencillo antes de ensuciarnos las manos. ¿Es una trampa de la percepción típica en la programación? Al considerar en conjunto estas miles de sutilezas, la suma de costos y tiempos para obtener un buen producto no es trivial.

El histórico de control de cambios de versiones está al final

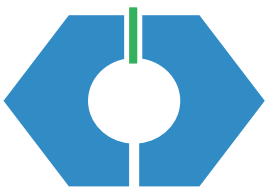
El software hecho a medida es programación que trata de adaptar herramientas pre-construidas para que se ajusten a la realidad de una empresa, ó en general a una situación específica.

El cliente puede participar en indicar algunas de sus preferencias, su visión de cómo debería ser el diseño, en indicar sus preocupaciones y las molestias que espera resolver con el desarrollo, en manifestar sus expectativas de manera clara en lo posible desde el comienzo del proyecto. No participa el cliente en todas las decisiones: existen algunos elementos que limitan la libertad absoluta del software a medida por causales de ética, de técnica, de legalidad, de impacto en el bienestar de las personas, de viabilidad, de alcance, de estándares de iure ó estándares de facto, entre otros.

Existen muchas modalidades, escuelas de software a medida, prácticas del mercado. Nosotros mismos hemos pasado por diferentes metodologías, y hemos podido depurar las prácticas que consideramos las mejores no sólo para el éxito de un negocio individual, sino también para la economía en general.

Las siguientes pautas buscan facilitar su comprensión acerca de por qué hemos tomado ciertas decisiones, para que comprenda usted que también sirven para su beneficio, y que por otra parte hay ciertas preocupaciones que son típicas pero que realmente no hacen la diferencia para su negocio, permitiendo que sea nuestra compañía la que se encargue de todo lo relacionado con la programación y los cuidados que ésta amerita.





Contenido de este documento:

¿Soy absolutamente libre de llevarme mis datos cuando quiera, para donde quiera, de tener copia permanente de los mismos?.....	2
¿Yo como cliente recibo el código de su trabajo?.....	2
¿Software a medida significa que yo puedo pedirles que hagan lo que yo quiera?.....	3
¿Software a medida significa que la propiedad intelectual pasa a ser mía?.....	4
¿El software a medida será instalado en mis computadores ó mis servidores?.....	5
No entiendo. Yo pago por un servicio a la medida, ¿no debería yo ser dueño de todo?.....	6
Es que yo no quiero estar amarrado a un único proveedor.....	6
Es que soy un planeador para una entidad del Estado, y no deseo contratos “intuitu personae”.....	8
Excepciones: ¿en qué situaciones podemos pensar de otra manera?.....	9
Historial de versiones de este documento.....	10

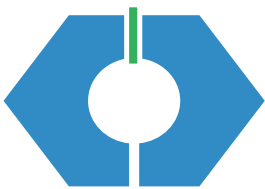
¿Soy absolutamente libre de llevarme mis datos cuando quiera, para donde quiera, de tener copia permanente de los mismos?

Sí. Los datos suyos son de su propiedad privada, y si en algún momento usted desea exportarlos para usarlos en otra parte, tiene el derecho a hacerlo. Esto no incluye la aplicación: el software convierte esos datos en información útil. La aplicación interpreta los datos, y ello se logra adaptando software pre-desarrollado a su situación específica.

¿Yo como cliente recibo el código de su trabajo?

NO, sería un alto costo adicional innecesario para su compañía y de todas formas no será útil para la gran mayoría de empresas que no se dedican exclusivamente al desarrollo de software. Veamos una metáfora: obviamente es un costo diferente el pagarle a un experimentado chef de cocina para que me prepare un plato, a pagarle un alto costo adicional para que me venda una experimentada y probada receta, donde dicha receta no sólo tiene en cuenta el proceso para lograr un buen sabor final, sino que el chef ha evolucionado detalles como saber mezclar productos para reducir el riesgo de personas alérgicas, tener en cuenta detalles médicos de los comensales para mezclar productos, considerar las mejores prácticas para comensales de colon irritable, identificar ingredientes ó productos con proveedores amigables con el medio ambiente y el buen trato animal, o sustituir unos productos por otros que no cumplan los mejores criterios. Es decir, si bien parece simple y fácil, actividades que desde afuera parecerían sencillos negocios, para un profesional están compuestos de centenares de





sutilezas que constituyen un producto completo y que, por esta suma de detalles, aumente confiabilidad por reducir inconvenientes y riesgos para usted como cliente, siendo además un producto más probablemente escalable y duradero que si fuera un producto improvisado con poca observación de las minucias.

Puntualicemos algunas razones:

1. Hablaré como el cliente. Si yo soy un cliente que solicita a un cocinero que elabore una comida con platos escogidos por mí, para unos invitados, el propósito es gozar de las mejores técnicas del equipo de cocineros para disfrutar de una cena, no es mi propósito hacerme a las recetas que los cocineros han evolucionado durante años profesionales. Lo segundo es mucho más costoso que lo primero, y no es relevante para el negocio esencial de su compañía, así que generalmente no será útil comprometer sus finanzas con una serie de altísimos gastos posteriores que no serán prácticos ni útiles para su compañía, que de todas formas tiene otro objeto social.
2. Seguridad informática: el software libre de código abierto es una muestra de los peligros de liberar el código fuente, en materia de vulnerabilidades informáticas por el hecho de que se conoce dicho código.

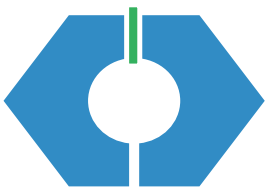
Imagine que usted está protegiendo un banco. No sería prudente que los delincuentes contaran con un mapa de ese banco que les permita conocer potenciales puntos de entrada no oficiales, corredores, sitios de bienes sensibles, zonas vigiladas, zonas grises con puntos ciegos no vigilados, y demás. Para la protección de su propia seguridad como usuario, ese código fuente (el mapa, en la metáfora) no debería ser público.

¿Software a medida significa que yo puedo pedirles que hagan lo que yo quiera?

No, hay varias causales de rechazo de una solicitud. Por una parte, usted sabe que todo trabajo tiene un alcance, que no es ilimitado. Pero existen otras causales de rechazo. Algunos criterios son:

- Conflictos éticos
- Respeto a los derechos de Habeas Data de las personas
- Que un desarrollo de software no represente un retroceso conceptual en ingeniería industrial, ó en operatividad.
- Legalidad de las solicitudes
- Solicitudes fuera del alcance de un proyecto, propuesta ó contrato.
- Que incluso dentro del alcance, se solicite un re-trabajo de algo ya realizado sobre lo cual no se objetó a tiempo.
- Que se considere que una decisión lo perjudica a usted ó a su compañía en el largo plazo, ó que





de manera ilegal ó desconsiderada se encuentre que se hace daño a alguien.

- Inconveniente relación costo/beneficio que perjudique un proyecto, ó su impacto sea mínimo en comparación con otros aspectos de mucha mayor importancia.
- Aspectos que son responsabilidad de terceras partes.

¿Software a medida significa que la propiedad intelectual pasa a ser mía?

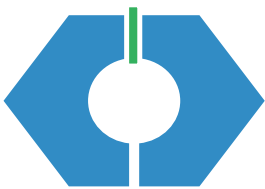
No. Regresemos al ejemplo del cocinero. Una idea brillante no representa nunca la totalidad de un esfuerzo que soporta por detrás el bagaje previo a su realización.

Si solicito a una empresa de cocineros que elabore una comida para alimentar con ciertos platos a un grupo de personas, ello no implica hacerme acreedor de las recetas que ese equipo humano ha elaborado y probado con los años. La empresa de cocineros me ofrece años de experiencia de platos que han sometido a prueba no sólo en técnicas de preparación, sino que también acumulan criterios para tomar decisiones velando por la salud de los comensales, cuidar de combinaciones peligrosas, selección de materias primas, productos y proveedores que participan en la elaboración, entre otro gran conjunto de secretos de procedimientos que de ninguna manera hacen parte de lo que un comensal ó contratante de los servicios de cocina debería pretender como entregable. Su propósito es darle un servicio de calidad (una comida, en el ejemplo), no se trata de entregar al cliente sus recetas.

Esta pregunta curiosamente ha surgido en algunas empresas en el pasado. Lo que se ha observado con experiencias en donde se entrega el código fuente, es:

- 1) El tener un código fuente no es garantía para usted de que un producto podrá ser transformado, ni le da realmente seguridad práctica, solamente una ilusión que pronto se muestra como inútil. De hecho, en Internet hay muchos paquetes de código libre que gozan de muchísima documentación, comunidades, e historia. Y ahí están, sin que sea trivial su transformación ó adaptación sin mencionar los riesgos de seguridad, incluso cuando se ubica a los pocos y más expertos programadores dispuestos a invertir tiempo de su vida en dichas transformaciones. Es algo que por supuesto hemos también probado en diferentes ocasiones para evolucionar a las ofertas que concebimos para usted como cliente.
- 2) Algunas tecnologías cuentan con la posibilidad de ingresar módulos externos como "plugins" que se programan extendiendo un núcleo funcional: pero surgen problemas como:
 1. La dificultad de garantizar la seguridad informática con esos plugins que tendrán acceso a los archivos y datos;
 2. Se ha observado que a través de plugins y plantillas de paquetes famosos de código abierto han ingresado vulnerabilidades y ataques informáticos de manera exitosa.



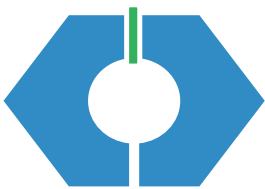


3. Existen vulnerabilidades informáticas publicadas de los paquetes comerciales, y sitios web tanto legítimos como ilegítimos en donde se encuentran colecciones de técnicas para "crackear" dichas vulnerabilidades.
4. La prueba de que esta amenaza es real, es revisar un log diario de diferentes sitios web de bajo tráfico: ya en diferentes microempresas pequeñas, en los logs de sus sitios web, se han detectado decenas de ataques diarios en donde robots automáticos intentan explotar vulnerabilidades pre-identificadas. Las empresas de hosting tienen clara esta realidad e incluso han decidido forzar actualizaciones automáticas de ciertos paquetes famosos de software, con lo cual el software externo reconocido por esas empresas se estará modificando de manera automática, introduciéndose no sólo los parches de seguridad, sino también nuevos puntos de fallo por estar alterando un sistema que ya podía llevar años en funcionamiento y pruebas.
5. Se ha detectado en algunas ocasiones falta de garantía de soporte por parte del fabricante y por parte de la empresa intermediaria de donde se haya descargado ó comprado el plugin ó la plantilla de extensión. Se ha observado que a la hora de buscar soporte ó responsabilidad por un "plugin" comprado para completar un software de código abierto, las empresas intermediarias no necesariamente se responsabilizan del error, y no siempre pueden ubicar al programador que la realizó.
6. Debido a que se trata de un único código fuente evolucionado en una empresa, se ha observado que incluso cuando una gran empresa cliente cuenta con los mejores programadores, por ejemplo en los bancos, al final tiene un único caso de uso. En esta lógica, cada equipo de programadores tiene que volver a re-descubrir los aspectos más básicos, sin que se acumule conocimiento, generando altos costos y tiempos para problemas que puede ya haber resuelto la competencia del mismo sector, y que de todas formas no son parte nuclear de su negocio, sino que simplemente deberían estar ya resueltos. Es una gran desventaja técnica y económica comparada con el caso opuesto, a saber: cuando una plataforma de base evoluciona con muchos estudios de caso.

¿El software a medida será instalado en mis computadores ó mis servidores?

No. Parte de las lecciones adquiridas con los años, es que todo software amerita cuidadosas decisiones técnicas sobre la infraestructura, para optimizar los recursos, agilizar los tiempos de reacción, brindar estabilidad y seguridad al cliente. Esto, sumado a los argumentos anteriores ya mencionados en otros puntos de este mismo documento. Adicionalmente, para que pueda aplicar la figura legal de computación en la nube por ejemplo para exclusiones de IVA, se deben cumplir





investigación
& programación

Invierta en investigación.
Genere posibilidades

ciertos criterios técnicos.

No entiendo. Yo pago por un servicio a la medida, ¿no debería yo ser dueño de todo?

El dinero que usted paga no es para realizar una plataforma desde ceros, sino que le ofrecemos una plataforma que hemos desarrollado y adaptado durante varios años a múltiples proyectos de diferente naturaleza, evolucionando con centenares de usuarios que la han puesto a prueba, recibiendo muchas observaciones y optimizándola para diversos fines. Así, antes de usted llegar, se han construido centenares ó miles de versiones, con decenas de miles de problemas resueltos, para ofrecer a usted una adaptación de nuestra plataforma. El empezar con una plataforma que cuenta con años de experiencia le brinda a su empresa mayor estabilidad, seguridad informática, y un acervo de situaciones sutiles que pueden presentarse y sobre las cuales hemos procurado robustecer cada vez más los productos que entregamos a nuestros clientes. “Eso es sencillo”, dirá usted, lo cual es un buen pensamiento a la hora de inventar en hombros de gigantes: pero no olvide que es una suma de centenas de miles de cuestiones sencillas.

Más aún, otra razón que se suma a esta tesis, es que en algunos de los desarrollos nuevos, algunas partes son cofinanciadas por nosotros y las otorgamos con un costo mínimo para usted, cuando observamos que es viable y justo, y sobretodo que no estamos perjudicando a otras personas.

Finalmente, observe usted que es nuestro propósito cooperar con el bien de un mercado en general que no sólo nos incluye a usted y a nosotros, sino a una economía en donde las decisiones equivocadas pueden perjudicar el desarrollo y la financiación del desarrollo y aprendizaje de la tecnología para otras personas. Es nuestra responsabilidad, suya y nuestra, tomar decisiones cuidadosas para las futuras generaciones.

Es que yo no quiero estar amarrado a un único proveedor

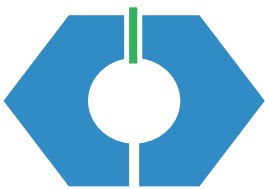
Aquí hay dos respuestas:

1. Sus datos son de su propiedad, y usted es libre de exportarlos cuando quiera, a donde quiera.
2. La aplicación tiene como tarea combinar esos datos e interpretarlos para usarlos como información. Usted es libre al tener el acceso a sus propios datos. Esta libertad no incluye el procesamiento de los mismos que realiza la aplicación.



Investigación y Programación S.A.S

Cel+Whatsapp: (+57) 312 571 89 85 - cel: (+57) 300 490 46 69 www.investigacionyprogramacion.com
contacto@investigacionyprogramacion.com - Bogotá- Chía, Cundinamarca, Colombia



3. Es más libre quien realiza acuerdos de largo plazo con un proveedor, que quien realiza acuerdos de largo plazo con una herramienta.

La palabra “amarrar” evoca una negativa falta de libertad. Pero en la práctica, el acuerdo con un proveedor significa tener un marco de trabajo que busca lograr una estructura de trabajo que permita agilizar soluciones. A veces perseguimos el tener muchas opciones por encima de tener mucha estabilidad y capacidad de ejecución. Y de todas formas las mejores opciones, si cumplen los criterios de viabilidad, se pueden construir con el proveedor de software con el cual usted tiene un acuerdo de largo plazo.

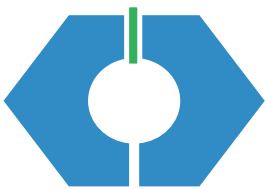
La lealtad a un proveedor es superior a la lealtad a una herramienta: en el primer caso (lealtad al proveedor), usted tiene la alternativa de cambiar ambos proveedor y herramienta, llevando sus datos a otra parte, pues son de su propiedad; pero en el segundo caso (lealtad a la herramienta) sin un único proveedor que se responsabilice por dicha herramienta, usted debe tantear con ambas cosas (tantear proveedor y tantear herramienta), mientras se distrae del propósito esencial de su negocio, que es diferente del desarrollo de software.

Quien está amarrado a una herramienta que no funciona ó no se puede escalar, es quien sufre, no quien realiza acuerdos de largo plazo con un proveedor.

Examinemos esta preocupación típica. Si el proveedor es justo con sus políticas y cumple un buen papel, procura máxima ética y cumplir buenos principios con sus clientes, el mantener relaciones de largo plazo no debería ser un problema. Pero supongamos que esto usted no lo sabe: usted conoce a una nueva empresa de desarrollo de software, y usted no sabe si este proveedor de software será ético, confiable y tendrá las mejores prácticas técnicas. Esa incertidumbre se compara, entonces, con la alternativa: si usted se organiza para tener varios proveedores que tengan acceso a su software y datos, entonces usted tendrá esa misma incertidumbre con cada nuevo proveedor que contrate, y tendrá que reproducir el mismo riesgo varias veces.

Cuando se posee un código fuente, se debe conseguir y financiar los honorarios de aquel programador dispuesto a realizar intervenciones. Se ha observado que en estas lógicas, ocurre que la empresa poseedora de un código fuente no contrata a un único programador sino que un código fuente empieza a tener participaciones de diferentes programadores con diferentes estilos de programación, técnicas, costumbres, llenando de “pequeños parches” el código, para soluciones “ad hoc”, que terminan estando desconectadas unas de otras, haciendo que el "caos" en el código crezca, y con él la dificultad de modificación. Esto genera que la empresa que por tener <<diferentes proveedores>>, es decir





<<diferentes programadores>> que manipulen el código, deba pagar crecientes sumas por modificaciones que serían más económicas por otros métodos, como es el de evolucionar un código a través de muchos proyectos por parte de una empresa de software dedicada a eso.

Vamos a suponer que al final, después de tantear y sufrir costos, pensando con el mayor optimismo, usted finalmente encontrará un proveedor que reúna en un solo proveedor un alto nivel ético y técnico, que respete la confidencialidad de su compañía, y que adicionalmente esté dispuesto a estudiar el código fuente para su compañía. <<Tenemos a la persona que reúne estas condiciones y está disponible>>, es su respuesta. Pero regresa usted entonces a una condición peor que su preocupación original. En efecto, ¿qué ocurrirá? Con el paso del tiempo, esta persona de excelentes cualidades humanas y técnicas, evolucionará versiones con UN código con UN caso de uso, que es el de su compañía. Al final, su empresa terminará dependiendo de ambas cosas únicas, programador y herramienta. En ese punto, su opción es exportar los datos a otra nueva herramienta.... Regresando al mismo punto original que ya hubiera tenido al contratar a un proveedor único de software a medida, que de todas formas le permite esa exportación de datos.

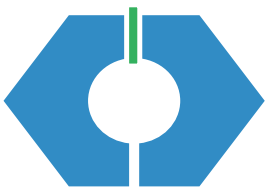
Recuerde que:

1. Usted no es el único que ha pagado por un desarrollo de software. Esto significa que el producto se soporta sobre una gran historia de proyectos y desarrollos. Y que recibirá servicio de soporte y sobretodo escalabilidad, por un proveedor experto en software, evitando a usted costos futuros de tantear soluciones ya probadas, reclutar nuevos programadores, y sobretodo reducir riesgos de errores por manipulaciones equivocadas.
2. La confiabilidad y estabilidad de un software a partir de cierto tamaño es algo que se logra con años de evolución de miles de correcciones, con centenas ó miles de usuarios puestos a prueba.

Pero hay excepciones si su proyecto es pequeño y específico, y por su tamaño es viable desarrollarlo desde ceros para su compañía. Esto se explicará en la sección de “excepciones”.

Es que soy un planeador para una entidad del Estado, y no deseo contratos “intuitu personae”





En la práctica, el riesgo es peor cuando no existe un marco de trabajo, y una entidad termina dependiendo de un programador ó un equipo de programadores que pueden tener rotación en cualquier momento. La responsabilidad de un trabajador oficial con la entidad estatal es sobre su función diaria, y es legal que ese trabajador renuncie ó sea liquidado, mientras que un proveedor de software tiene responsabilidad con el buen rendimiento de la aplicación en el tiempo. Los contratos y responsabilidades son diferentes.

De hecho, citemos un caso concreto en donde conocimos exactamente esa preocupación. Con regalías de la nación se financió la construcción de un software para promocionar el turismo. La financiación no incluía mantenimiento ni soporte en el largo plazo, únicamente la construcción del desarrollo. No había en el Estado nadie encargado de velar por conservar ese resultado en el tiempo. El software no requería mantenimiento, así que no ameritaba tener un ingeniero de planta velando por él. Pero cuando hubo problemas, el grupo receptor del producto no tenía los conocimientos ni acceso para resolver los problemas. Y el producto finalmente se perdió en el tiempo, por falta de un marco y un responsable claro para el soporte y escalabilidad. El exceso de opciones generó la muerte del proyecto. ¿A que usted conoce casos similares en entidades del Estado, respetado lector?

En otro caso conocido, de una gran entidad, se decidió realizar “soporte inhouse” de un software de alto tráfico para nivel nacional. ¿El resultado? Solicitamos a la entidad sus cuentas de cuánto dinero habían gastado: nos reportaron con sus cuentas oficiales, que en menos de 15 años habían gastado más de \$100.000.000.000 (cien mil millones de pesos colombianos, es decir más de U\$33 mil millones de dólares para esa fecha), cifra creciente año tras año.

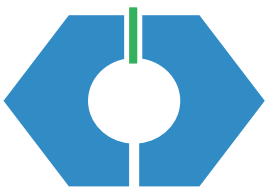
Excepciones: ¿en qué situaciones podemos pensar de otra manera?

Sí tiene excepciones. Los casos usuales que hemos mencionado en otras secciones describen situaciones maduras en donde un código lleva años de evolución y tienen un cierto tamaño, pero no son el único caso posible en Desarrollo de Software A Medida.

A veces nos llaman competidores de desarrollo ó empresas del sector de TICs, ó empresas de ingeniería, que necesitan que les desarrollemos un plugin muy específico, ó un API para su pre-existente plataforma, y requieren el código porque es su herramienta de trabajo, ó porque tienen un departamento de desarrollo que desea aprovecharlo, y lo solicitan explícitamente.

En general, si el proyecto es muy pequeño (menos de 2000 líneas de código), que no lleve años de





investigación
& programación

Invierta en investigación.
Genere posibilidades

maduración en múltiples casos, sino que se trata de una adaptación muy concreta en donde no usamos de nuestras plataformas, sino que se trata de un código desarrollado desde ceros para una situación muy específica de su compañía y exclusivamente para la misma, bien puede considerarse una excepción, si su compañía manifiesta explícitamente, desde antes de negociar, el estar dispuesta a pagar el excedente del código fuente. Estos son casos muy particulares y no son lo usual en el desarrollo de software. Por ejemplo, si usted ya tiene paquetes de software, si el desarrollo consiste en unir dos paquetes pre-existentes, y nuestra labor es únicamente la tarea de unión de esos paquetes, naturalmente sí es viable negociar el código a un precio razonable.

En tales excepciones, el acuerdo debe ser respetuoso de la ley 23 de 1982 y demás legislaciones ó decisiones multilaterales relevantes, para garantizar que se fomente en general el respeto a los derechos de autor, el derecho a la reivindicación de paternidad de las obras, y que si existe transferencia patrimonial (no moral) de derechos de autor, ello se realice juiciosamente, diferenciando en lo posible las modalidades contempladas en la ley.

Historial de versiones de este documento

Versión 0.1: Noviembre 01 de 2019

Versión 0.2: Diciembre 04 de 2019, se aclararon algunos párrafos, se corrigió redacción. <<Las prácticas que consideramos mejores (...)>>, y por qué el cliente en general no debe asumir el innecesario costo del código fuente. Se agregó la sección de “excepciones” para proyectos muy pequeños y específicos.



Investigación y Programación S.A.S

Cel+Whatsapp: (+57) 312 571 89 85 - cel: (+57) 300 490 46 69 www.investigacionyprogramacion.com
contacto@investigacionyprogramacion.com - Bogotá- Chía, Cundinamarca, Colombia