

Offshore

Luis García Estrades

December 13, 2014

1 INTRODUCCIÓN

La subcontratación de software en el extranjero (en adelante “offshoring”) se ha vuelto muy popular en los últimos años. Los beneficios son fácilmente entendibles cuando comparas el salario de un ingeniero de Estados Unidos con el de la India. En este documento se da respuesta a las siguientes preguntas: ¿Merece la pena el offshoring? ¿Son los ahorros tan cuantiosos como parece? ¿Se ve comprometida la calidad del software o se mejora? ¿Hay riesgos asociados con el offshoring? ¿Hay maneras de cubrir estos riesgos? ¿Hay tipos de software que son mejores a la hora de hacer offshoring? ¿Cómo reconocer los indicadores de que un proyecto de offshoring empieza a fallar? ¿Cuáles son las alternativas al offshoring?

2 VENTAJAS

¿Cuáles son las principales ventajas del offshoring?

La que más llama la atención es el ahorro de costes debido a la mano de obra barata.

Hay abundantes desarrolladores con un perfil técnico muy bueno y con ganas de trabajar.

Algunas empresas se inclinan hacia el offshoring debido al potencial ahorro en impuestos ya que pueden explotar las políticas del gobierno Indio en cuanto a los impuestos, incluyen diez años libres de impuestos para empresas de I+D centradas en la investigación industrial y una reducción del 150% para investigación científica y en costes de desarrollo.

Algunas empresas expresan que el offshoring les permite centrarse en operaciones financieras críticas mientras que la gestión del proyecto y el software no tan crítico recae en manos de las empresas extranjeras.

Algunas empresas han reportado que la calidad del software extranjero es superior al que podrían haber llegado a desarrollar internamente.

Algunas empresas utilizan el offshoring para desarrollar el software casi las 24 horas del día ya que cuando un equipo de desarrolladores en Estados Unidos acaba su jornada laboral, un equipo de desarrolladores en la India lo retoma justo donde lo han dejado, mejorando los tiempos de respuesta frente al mercado.

3 INCONVENIENTES

Todo tiene su parte negativa y el offshoring también está incluido, es importante tener en mente los inconvenientes del offshoring antes de enviar software al extranjero.

El inconveniente más obvio es el idioma, las diferencias en el idioma pueden acarrear problemas de comunicación y una buena comunicación entre el usuario y el desarrollador es clave para extraer con precisión los requisitos del software a desarrollar.

Los viajes y costes de comunicación pueden ser substanciales si el partner permanece por largos períodos de tiempo en el extranjero.

Las diferencias tecnológicas pueden causar problemas, es importante asegurarse de que las plataformas de desarrollo del software sean compatibles con las plataformas del producto final.

Algunos otros factores como diferencias de estándares, prácticas de desarrollo de código, estilos de documentación y ética en el código pueden llegar a ser problemáticos.

4 CONSIGUIENDO LA CALIDAD

Con el posible potencial para todos estos problemas, ¿por qué empresas estadounidenses utilizan el offshoring como solución? Muchas dicen que el código obtenido no solo es más barato sino que también es de mayor calidad. Las empresas que desarrollan software en la India se han dado cuenta del papel que juega la calidad en su modelo de negocio. Un factor que cuantifica la calidad del software es el rating CMM (Engineering Institute's Capability Maturity Model) cuyo mayor nivel es el número 5 y la mitad de las organizaciones evaluadas según el CMM del mundo con este nivel de calidad se encuentran en la India.

Muchas empresas estadounidenses creen que resultaría demasiado costoso que el departamento de desarrollo llegara a este nivel de calidad y por ello creen que es más barato utilizar el offshoring en lugar de optimizar sus procesos en Estados Unidos. Visualizan sus mejores oportunidades o externalizando software a alguna de estas empresas que operan a este nivel o empezar un nuevo grupo interno de desarrollo que siguen controles de calidad desde el principio.

5 INEXISTENCIA DE GARANTÍAS DE CALIDAD

No necesariamente externalizar es una solución adecuada a la hora de desarrollar software de calidad ya que un nivel 5 de CMM no garantiza siempre obtener código de calidad ya que no todas las clasificaciones se realizan de una manera estrictamente controlada y es importante saber quién ha realizado dicha clasificación.

Una compañía puede sobreestimar su rating de CMM si se evalúa internamente y puesto que el ratio CMM no es un certificado como tal, no existen requisitos de actualización del rating y éste puede ser que esté desactualizado. Otro factor que puede llegar a engaños es el porcentaje de la empresa que se ha evaluado ya que puede ser que se haya evaluado una pequeña parte y el ratio CMM puede dar a pensar que es sobre toda la empresa que se contrata.

6 RIESGOS

Siempre existen riesgos de realizar offshoring para una parte del software y a parte de no recibir la calidad esperada existen otros riesgos a considerar.

Se deberían considerar los problemas jurídicos ya que a menudo las leyes extranjeras son inadecuadas a la hora de proteger la propiedad intelectual ya que no hay una ley que proteja las compañías estadounidenses. Considerando que hay leyes estadounidenses de protección de datos que no se aplican al extranjero, el riesgo de que las empresas extranjeras vendan el código a la competencia claramente existe.

Se han reportado casos en que se ha externalizado código y después de desarrollarlo, la empresa india lo entregó al cliente y utilizaron el software para crear una versión en la India del producto del cliente.

A pesar de estos robos de propiedad intelectual cabe resaltar que la India tiene mejores entornos culturales y jurídicos en cuanto a propiedad intelectual que otros destinos de offshoring como Rusia y China.

7 CANDIDATOS PARA EL OFFSHORING

Analizando los posibles riesgos de externalizar software se puede determinar las clases de software que son más apropiados para externalizar, entre ellos: diseño y desarrollo web, proyectos e-commerce que tienen una parte externa que se encarga de la seguridad, mantenimiento remoto y mejoras de features.

En contraposición, los peores candidatos son: desarrollo de software de industrias regularizadas, software perteneciente a sectores militares, software del gobierno, software que relacionado con leyes de diferentes países, proyectos tecnológicamente muy complejos debidos al volumen de comunicación requerido.

8 INDICADORES DE FRACASO

Darse cuenta de los riesgos involucrados en la externalización de software nos permite reconocer indicadores de que el desarrollo del código no está yendo como debería. Entre estos indicadores se encuentran: descripciones no detalladas en las tareas del planning del proyecto, estimaciones de tiempo que parecen imposibles, incoherencia entre documentos y software, informes de progreso esporádicos o inexistentes, desconocimiento de los desarrolladores del proyecto, pobre calidad del trabajo, demasiada dependencia en el trabajo subcontratado, etc.

9 RECONSIDERANDO LA EXTERNALIZACIÓN DE SOFTWARE

Antes de externalizar el desarrollo del software miles de kilómetros una empresa debería evaluar los riesgos y beneficios de hacerlo, el mayor beneficio suele ser el ahorro del coste de desarrollarlo de forma interna.

Por otro lado no hay razones para que empresas norteamericanas no realicen código de calidad y reducir el coste a través de una buena eficiencia. Hay bastantes desarrolladores que siguen estrictas guías de calidad, consiguen incrementar la productividad y no presentan ninguno de los inconvenientes del offshoring.

10 BUENOS MÉTODOS

Una vez que se ha hecho la contabilidad, se han analizado los riesgos, si la evaluación indica que hay significantes ventajas para externalizar el software, entonces es hora de pensar cuál es la mejor manera de hacerlo. Hay que cubrir los riesgos concernientes a la propiedad intelectual: solo externalizar a países con remedios jurídicos si el robo de propiedad intelectual ocurre, indicar cláusulas de confidencialidad en los contratos, indicar en el contrato el propietario del código que se va a desarrollar, tener constancia por escrito de de dónde proviene el código, mantener el software crítico o central de la empresa dentro de la empresa (no externalizar partes críticas de la empresa), controlar quién puede acceder a cualquier documento o software creado por la empresa y por último controlar la distribución del software creado.