

Guía de supervivencia a la resolución en origen para probadores

Paul Gerrard, un gurú de las pruebas y consultor, analiza en una serie de artículos una gran variedad de cuestiones relacionadas con las pruebas. La resolución en origen consiste principalmente en adelantar la planificación de las pruebas a una fase más temprana del proceso. En este artículo, Paul habla sobre lo que está impulsando la transición a la resolución en origen, sobre cómo está cambiando el papel de las pruebas y cómo le afectará todo esto en un mundo más ágil.

Paul Gerrard
Gerrard Consulting

Con el patrocinio de



Guía de supervivencia a la resolución en origen para probadores

Contexto

Hace unos cinco años, acuñé el término “redistribución de las pruebas” (*redistributed testing*) para describir un cambio en el mundo de las pruebas. Según este cambio, los usuarios, analistas, desarrolladores y probadores se redistribuirían de forma responsable para las pruebas y colaborarían de forma más eficaz. Dicho cambio implicaría adelantar las actividades de pruebas (y, posiblemente, también las responsabilidades). El término utilizado con el que se suele hacer referencia a este método es “resolver en origen”.

La resolución en origen puede conllevar que los desarrolladores asuman una mayor propiedad y responsabilidad de sus propias pruebas; así como que los probadores intervengan antes, cuestionen los requisitos y aporten ejemplos a los desarrolladores mediante un proceso de desarrollo basado en el comportamiento (BDD, *behaviour driven development*). También puede implicar que los usuarios y los analistas empresariales asuman una responsabilidad total de las pruebas junto con los desarrolladores, lo cual puede implicar que no haya ningún equipo de pruebas ni probadores. Hemos sido testigos de todo tipo de configuraciones y, por supuesto, no existe “el único método adecuado”.

Este artículo describe cómo está cambiando el papel de las pruebas y cómo le afectará todo esto en un mundo más ágil.

La resolución en origen no es nada nuevo

Durante todo el tiempo que he estado en el mundo de las pruebas (desde 1992) y, sin lugar a dudas, ya desde antes, los defensores de las pruebas han predicado el mantra “haz pruebas pronto y con frecuencia”. El modelo en W,¹ introducido por mi amigo y colega Paul Herzlich ya en 1993 sugería que todos los artefactos de un proceso dispuesto en etapas (tanto documentales como de *software*) pueden probarse (y deberían probarse con frecuencia).

Aunque por aquel entonces el modelo en cascada era el método de ciclo de vida dominante, el número o la duración de las etapas no son lo más importante. El principio subyacente era que *las fuentes de conocimiento que aportan indicaciones para el diseño y el desarrollo de software deben cuestionarse o someterse a pruebas*. En un proyecto en etapas, esto podría conllevar revisiones formales. En un proyecto ágil, el probador (o el desarrollador, el analista empresarial o el usuario) puede sugerir escenarios (o ejemplos) que cuestionen al autor de un requisito o un caso práctico para estudiar detenidamente ejemplos concretos y debatirlos antes de escribir el código.

La resolución en origen consiste principalmente en adelantar la planificación de las pruebas a una fase más temprana del proceso.

¿Significa eso que todo lo que requiere la resolución en origen es que los probadores participen antes y planteen preguntas incómodas? ¿De verdad es algo tan sencillo? Bueno, no exactamente.

¿Qué impulsa la resolución en origen?

El mercado se está viendo afectado por varios cambios, que están impulsando nuevos comportamientos en nuestro sector. Algunos empezaron a producirse hará unos cinco años, pero otros son más recientes. Ahora bien, ¿qué quiere decir eso de que “empezaron a producirse”? Significa que hubo suficientes personas que promocionaron nuevos métodos cuya adecuación quedó demostrada y fueron lo suficientemente creíbles para que otras personas los adoptaran. Estos métodos “atravesaron el abismo”, por utilizar las palabras de Geoffrey Moore,² y se hicieron viables para todo el negocio y la comunidad informática.

Estos son los principales cambios que intervinieron en los fenómenos de la resolución en origen:

1. El enfoque del desarrollo basado en el comportamiento ha permitido la colaboración entre los desarrolladores, los usuarios o analistas empresariales y los probadores mediante casos empresariales. El desarrollo basado en el comportamiento goza actualmente de una mayor adopción porque fomenta una colaboración más adecuada en los equipos ágiles.

2. El concepto de la entrega continua³ también ha estado entre nosotros desde hace unos 5-10 años, y sus raíces se encuentran en los enfoques de automatización de las compilaciones y los lanzamientos muy automatizados preconizados por las grandes empresas en línea. En la actualidad, la adoptan la mayoría de las organizaciones con presencia en Internet.
3. La entrega continua sistematizó y aceleró el proceso de lanzamiento mediante la automatización. No obstante, también puso de relieve los retrasos en la implementación de la producción y el cambio en la infraestructura que habían enmascarado anteriormente los procesos compilación, pruebas y lanzamiento lentos. DevOps es un cambio cultural y de mentalidad por el que los desarrolladores colaboran de una forma mucho más estrecha, particularmente con el personal de operaciones. Actualmente, aparecen nuevas herramientas prácticamente cada día y los distribuidores promocionan DevOps como “el próximo gran éxito”. Es una situación muy emocionante y dinámica.
4. Las plataformas sociales, móviles, de análisis y en la nube representan un cambio en la forma en la que las organizaciones gestionan su negocio y los cambios en los sistemas en el espacio móvil. La experimentación en las empresas, implementada como cambios en los sistemas de producción, se monitoriza en detalle. Se procesan los “grandes datos” capturados y se toman decisiones a partir de los análisis obtenidos.

La experimentación frecuente con sistemas de producción posibilita la innovación empresarial “a la velocidad del marketing”. La experimentación se encuentra en el corazón de lo que parece ser la tendencia más importante de la segunda década de este siglo: la “transformación digital”. La transformación digital o, simplemente “lo digital”, está captando la mayor parte de la atención (y el presupuesto) ahora mismo. Los comerciales prometen a los clientes un acceso más rápido y de mayor calidad por medio de un mayor número de canales (la mayoría móviles).

En mi artículo “Digital Transformation, Testing and Automation”⁴ describo la revolución digital y propongo algunas respuestas, por lo que podría interesarle.

¿Qué implicaciones tiene la resolución en origen para los probadores?

La resolución en origen implica que, siempre que sea posible, se debe aportar retroalimentación que ayude al equipo a comprender, cuestionar y mejorar los objetivos, los requisitos, el diseño o la implementación. A muchos probadores les surge hacer esto de forma natural, pero no a todos. Los usuarios, los analistas empresariales, los desarrolladores y todo el equipo deben estar preparados para proporcionar y aceptar la retroalimentación de este modo. Puede haber resistencia a hacerlo, pero el objetivo general es ejecutar un proyecto de una forma más adecuada y fundamentada, ni más ni menos.

¿Qué papel desempeña el probador en el mundo de las pruebas de resolución en origen? Bueno, la forma más sencilla de resumir este comportamiento es “intervenir pronto”, cuanto antes. El probador debe participar en el debate y colaborar en las ideas, los requisitos y en todas las etapas cuyo resultado aporte valor a la entrega final del proyecto. Para simplificar, el probador debe cuestionar las fuentes de conocimiento, ya sean partes implicadas, usuarios, desarrolladores, casos empresariales, documentos o conocimientos recibidos.

El enfoque más habitual consiste en “cuestionar mediante ejemplos”. En todas las etapas, se deben considerar estos ejemplos como pruebas. Es posible que se descarten rápidamente tras su uso o que se codifiquen en la automatización de pruebas o en las comprobaciones manuales. Estos ejemplos podrían utilizarse estratégicamente para señalar defectos en el planteamiento del personal u ofrecerse a los desarrolladores como ideas o semillas para pruebas de desarrolladores, por poner unos ejemplos. También podrían utilizarse como ayuda de asesoramiento para que los usuarios o los desarrolladores puedan ver más fácilmente cómo crear pruebas más adecuadas.

Los proyectos de *software* se han descrito como un proceso de adquisición de conocimientos.⁴ Estos conocimientos se reúnen a lo largo del proyecto y suelen evolucionar con el paso del tiempo. El objetivo de la resolución en origen es garantizar estos conocimientos mediante el cuestionamiento y la realización de pruebas cerca del origen y, siempre que sea posible, asegurarse de que estas sean fiables antes de que se congelen en el código.

La resolución en origen va más allá de la filosofía de priorizar las pruebas. La metodología ágil siempre ha fomentado la colaboración y la retroalimentación rápida, por lo que la resolución en origen podría considerarse simplemente como el enfoque de retroalimentación rápida definitivo.

Si la adopta, la resolución en origen tendrá un efecto profundo sobre su forma de trabajar como probador.

¿Cómo debe un probador aplicar la resolución en origen?

Parece que la resolución en origen no es solo una moda pasajera, sino que además, pronto le afectará. ¿Cómo le afectará si trabaja en un equipo de pruebas de sistemas? Si forma parte de un equipo ágil, ¿continúa teniendo importancia?

¿Qué debe hacer?

Ya hace tiempo que defendemos el enfoque de la resolución en origen como un elemento fundamental de la estrategia de las pruebas para proyectos ágiles. En un contexto ágil, la estrategia de las pruebas se puede contemplar como una serie de “intervenciones ágiles”. Hay momentos críticos en todos los proyectos en los que las oportunidades para reunir y ofrecer retroalimentación se presentan por sí mismas. El probador debe centrarse en estos momentos críticos y estar preparado para realizar su contribución en cuanto se produzcan.

Presenté el planteamiento que respalda este método hace poco en un seminario web⁶ y utilicé un caso práctico de un cliente para ilustrar dónde podrían producirse estas intervenciones. En sus propios proyectos, debe identificar los “momentos críticos” y las decisiones que usted y su equipo pueden tomar. Por ejemplo, en el caso de que escriba pruebas unitarias para desarrolladores, debe proporcionarles ejemplos para que se pongan manos a la obra, ¿o acaso debería asesorarlos para mejorar su capacidad de realización de pruebas?

Su papel cambiará con casi toda probabilidad. Es posible que los probadores no solo estén adelantando la planificación, sino que estén resolviendo en origen, y que usted se convierta en el sirviente de pruebas de los desarrolladores. Probablemente, este no sea el mejor resultado para usted en el proyecto. Le recomendamos que identifique los momentos críticos, proponga su contribución y negocie con su equipo. Debe ofrecer más liderazgo y orientación en las pruebas, en lugar de prestarse voluntario para limitarse a asumir la responsabilidad por el trabajo de pruebas. Si adopta este enfoque, le resultará mucho más sencillo demostrar su valía ante el equipo, ya que este no necesitará tantos probadores.

Soy un director/gestor de pruebas. ¿Qué debo hacer?

Si actualmente desempeña las funciones de gestor o director de pruebas, puede que le resulte más complicado justificar su cargo si el propósito de la gestión es reducir el coste de las pruebas resolviendo en origen. Si su organización está avanzando en esta dirección, probablemente tendrá que tomar una decisión determinante para su carrera profesional. ¿Dónde desea estar dentro de cinco años? ¿O dentro de seis meses? Hemos identificado cinco opciones generales que podrían presentarse.

1. **Poner sus habilidades con las pruebas y el control de calidad al servicio de las empresas:** Si asciende en la pirámide jerárquica en dirección a las partes implicadas, su papel podría ser el de asesorar a los líderes empresariales que deseen tomar el control de los proyectos informáticos. Como agente independiente, comprendería las preocupaciones de las empresas y las comunicaría en los proyectos. Asesoraría e influiría en la dirección del proyecto, revisaría el rendimiento y los logros, interpretaría los resultados y asesoraría a las partes implicadas.
2. **Gestionar los requisitos de conocimientos:** En esta función, controlaría los conocimientos necesarios para definir y crear sistemas. Si utiliza sus destrezas más importantes, garantizará la claridad y la precisión de los requisitos y los ejemplos que ilustran las funciones que se utilizan. Ayudaría a la empresa y a los desarrolladores a decidir cuándo se puede confiar en los requisitos hasta el punto de que se pueda crear y probar el *software* de forma razonable. Gestionaría los requisitos, así como el glosario y el diccionario de uso de conceptos empresariales y elementos de datos. Prestaría un servicio de análisis que repercutiría positivamente en la empresa.
3. **Ser un maestro de las pruebas** (prestar una función de control a los equipos, los proyectos y las partes implicadas): Esta es una función parecida a la primera que hemos indicado, pero para entornos más orientados a metodologías ágiles. Ejercería como un profesional especializado en las pruebas y el control de calidad encargado de mantener los proyectos ágiles fieles a sus principios. Trabajaría codo con codo con clientes y propietarios de productos en sus propias instalaciones. Contribuiría a que los proyectos reconocieran el riesgo y reaccionaran en consecuencia, asesoraría e instruiría al equipo y gestionaría sus actividades de pruebas y, de vez en cuando, también las ejecutaría.

4. **Ser un maestro de DevOps:** Gestionar el flujo de información vital en todos los sentidos de los procesos de DevOps (procesos automatizados de creación, pruebas e implementación). Este flujo de información es esencial. Quizá podría definir y supervisar los procesos utilizados para gestionar los flujos que permiten controlar el cambio, las pruebas y la entrega.
5. **Gestionar equipos externos o en otras ubicaciones:** En este caso, dejaría su equipo de pruebas local y gestionaría la transferencia de trabajo a proveedores externos o en otras ubicaciones. Sería un experto en el flujo de información y gestionaría las relaciones con el equipo de pruebas externo, monitorizaría su rendimiento y garantizaría sus resultados.

Si todavía no se le ha trasladado a la resolución en origen, debería echar un vistazo tanto dentro como fuera del equipo y pensar en cómo podría cambiar su posición en un futuro cercano. Su cargo acabará cambiando, pero debería tener un cierto margen de elección sobre su evolución. Le deseo suerte en las elecciones que haga.

Acerca del autor

Paul Gerrard es consultor, profesor, escritor, *webmaster*, desarrollador, probador, conferenciante, entrenador de remo y editor. Ha realizado tareas de consultoría en todos los aspectos de las pruebas del *software* y el control de calidad, y se ha especializado en el control de las pruebas. Ha presentado tutoriales y charlas destacados en congresos sobre pruebas en toda Europa, EE. UU., Australia, Sudáfrica y, en ocasiones, ha sido galardonado por ello.

Formado en la Universidad de Oxford y el Imperial College de Londres, Paul ganó en 2010 el premio Eurostar European Testing Excellence Award y, en 2013, fue galardonado con el European Software Testing Award (TESTA) en reconocimiento de su carrera profesional.

En 2002, Paul escribió el libro *Risk-Based E-Business Testing* con Neil Thompson. Posteriormente, escribió *The Tester's Pocketbook* en 2009. Paul fue coautor del libro *The Business Story Pocketbook* con Susan Windsor en 2011 y escribió *Lean Python* en 2014.

En 2014, Paul fue el Director de Programa del congreso EuroSTAR Conference que tuvo lugar en Dublín.

Es el Director de Gerrard Consulting Limited, de TestOpera Limited y anfitrión del Test Management Forum.

Correo electrónico: paul@gerrardconsulting.com

Twitter: @paul_gerrard

Página web: gerrardconsulting.com

Para obtener más información, visite **Desarrollo y pruebas** con CA Technologies.



Comuníquese con CA Technologies en ca.com/es



CA Technologies (NASDAQ: CA) crea *software* que impulsa la transformación de las empresas y les permite aprovechar las oportunidades que brinda la economía de las aplicaciones. El *software* se encuentra en el corazón de cada empresa, sea cual sea su sector. Desde la planificación hasta la gestión y la seguridad, pasando por el desarrollo, CA trabaja con empresas de todo el mundo para cambiar la forma en que vivimos, realizamos transacciones y nos comunicamos, ya sea a través de la nube pública, la nube privada, plataformas móviles, entornos de *mainframe* o entornos distribuidos. Para obtener más información, visite ca.com/es.

Referencias

1. "The W-Model", <http://blog.gerrardconsulting.com/?q=node/531>
2. "Crossing the Chasm" y otros libros de Geoffrey A Moore, <http://www.chasmstitute.com/>
3. La definición de "entrega continua" de Martin Fowler, <http://martinfowler.com/bliki/ContinuousDelivery.html>
4. "Digital Transformation, Testing and Automation", blog de Paul Gerrard, <http://blog.gerrardconsulting.com/?q=node/660>
5. "The Laws of Software Process", Philip G Armour
6. Seminario web: "Agile Test Strategy", Paul Gerrard, <http://blog.gerrardconsulting.com/?q=node/627>