

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
 Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
 HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
 Gestor de contenidos (Alfresco)
 Aplicaciones híbridas

Tareas programadas (Quartz)
 Gestor documental (Alfresco)
 Inversión de control (Spring)

Control de autenticación y
 acceso (Spring Security)
 UDDI
 Web Services
 Rest Services
 Social SSO
 SSO (Cas)

JPA-Hibernate, MyBatis
 Motor de búsqueda empresarial (Solr)
 ETL (Talend)

Dirección de Proyectos Informáticos.
 Metodologías ágiles
 Patrones de diseño
 TDD

BPM (jBPM o Bonita)
 Generación de informes (JasperReport)
 ESB (Open ESB)

AdictosAlTrabajo

Final de Terrakas
¡¡Ven al estreno!!
terrakas.com



autentia
Soporte a desarrollo informático
Hosting patrocinado por
enREDados

Entra en Adictos a través de  

E-mail

Contraseña

Entrar [Deseo registrarme](#)
[Olvidé mi contraseña](#)



[Inicio](#) [Quiénes somos](#) [Formación](#) [Comparador de salarios](#) [Nuestros libros](#) [Más](#)

» Estás en: [Inicio](#) [Tutoriales](#) [Control de la calidad, aseguramiento de la calidad y calidad total en el de...](#)



Roberto Canales Mora

Creador y propietario de AdictosAlTrabajo.com, Director General de Autentia S.L., Ingeniero Técnico de Telecomunicaciones y Executive MBA por el Instituto de Empresa 2007.

Twitter: [Seguir a @rcanalesmora](#) 1,232 seguidores

Autor del Libro: [Informática profesional, las reglas no escritas para triunfar en la empresa](#)

Puedes consultar mi CV y alguna de mis primeras aplicaciones (de los 90) [aquí](#)



[Ver todos los tutoriales del autor](#)

...me gusta prestar a
gente de mi provincia...



Antonio, usuario de
comunitae



Fecha de publicación del tutorial: 2013-04-25

Tutorial visitado 169 veces [Descargar en PDF](#)

Control de la calidad, aseguramiento de la calidad y calidad total en el desarrollo de software

0. Índice de contenidos.

- 1. Introducción
- 2. Control de calidad
- 3. Aseguramiento de la calidad
- 4. Mejora continua y calidad total

1. Introducción

Ayer estuve hablando en el evento XPDday y, como sólo disponía de media hora, y me enrolló con facilidad, no pude explicar como quería algunas de las transparencias y aprovecho ahora para hacerlo.

2. Control de calidad

Cuando se construye un proyecto de lo que sea, aunque centrámonos con el software, podemos comprobar la calidad al final del proyecto. Parece evidente que si se encuentra un error y ya hay plazos comprometidos con terceras partes (como campañas de marketing), si hay algún error va a ser complicado poder reaccionar. Estamos acostumbrados a zafarranchos de última hora con decenas de personas haciendo un sobre-esfuerzo y sacrificando la calidad...

Catálogo de servicios Autentia



Síguenos a través de:



Últimas Noticias

» [Atención, APLAZADO Estreno último capítulo de Terrakas](#)

» [Vendedor: Soy inseguro, filtra o elige por mí: si quieres que te compre.](#)

» [Comentando el libro: El arte de pensar, de Rolf Dobelli](#)

» [Ya está a la venta mi segundo libro: Planifica tu éxito, de aprendiz a empresario](#)

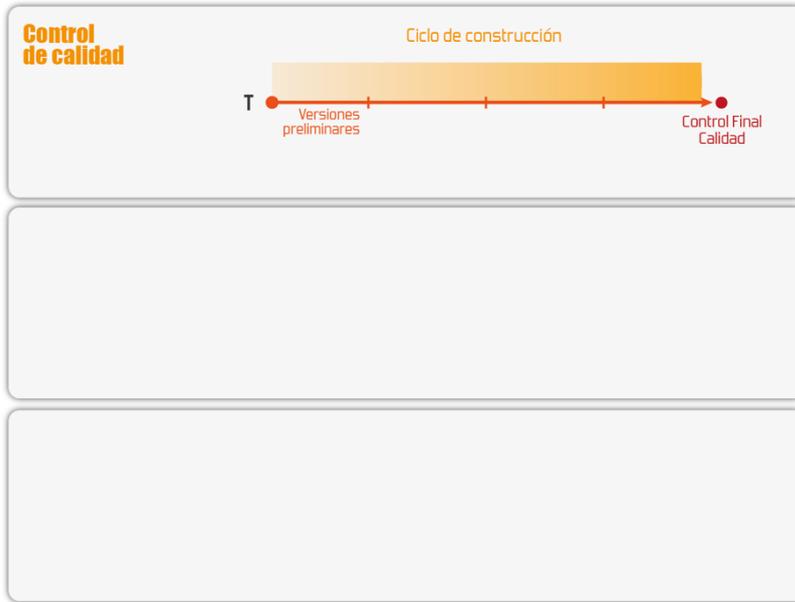
» [Ya esta disponible en eBook mi primer libro: Informática Profesional](#)

[Histórico de noticias](#)

Últimos Tutoriales

» [Uso de Requirejs para modularizar una App creada con Emberjs](#)

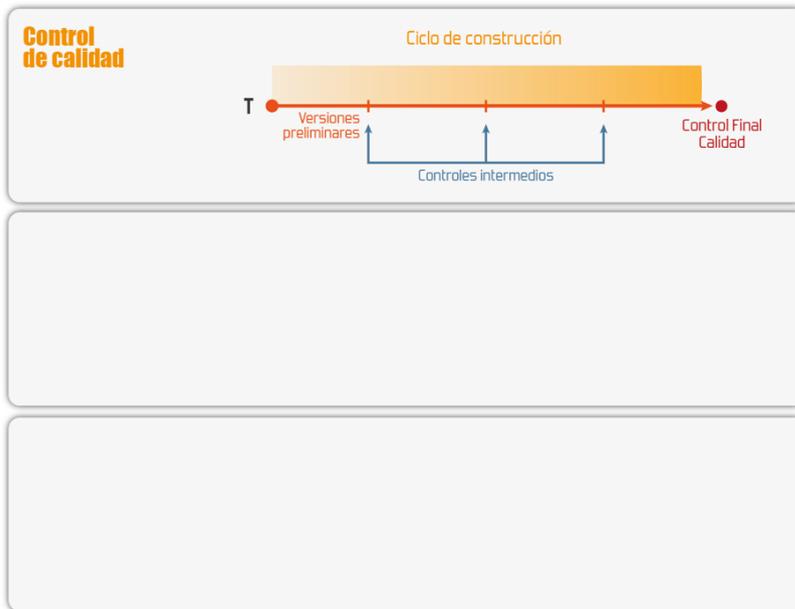
» [Instalación de Redmine \(Bitnami\) e integración con Subversion.](#)



Medidor de calidad



Para evitar sorpresas al final del proyecto se podrían establecer controles de calidad en puntos intermedios. Mejor todavía, si se hace en bases regulares. Esto, al menos, ayuda a detectar carencias formativas importantes.



Medidor de calidad



Es de esperar que deberíamos controlar aquellos puntos que previamente hemos definido. Imagina que subcontratamos un proyecto sin especificar previamente los criterios de aceptación y luego sorprendemos al proveedor con miles de condiciones... eso provocará un conflicto. Por tanto, deberíamos diferenciar en nuestro vocabulario el concepto "planificación de la calidad" y "control de calidad"

3. Control de calidad

El control de la calidad al final, por si mismo, parece insuficiente. Parece lógico que el propio proceso de construcción de software debería disponer de unos procedimientos que garanticen que la calidad está integrada, lo que vamos a llamar medidas de "aseguramiento de la calidad".

- » [Introducción a RequireJS](#)
- » [Conexión con mysql desde iSeries](#)
- » [Descubriendo Responsive Web Design](#)

Últimos Tutoriales del Autor

- » [Reactívatelo dinosaurio!!!](#)
- » [Nos han cambiado el agua en el sector informático](#)
- » [Modela tu mercado en base a la demanda y no a la oferta.](#)
- » [Mirar el todo: llevar el agilismo a las grandes organizaciones.](#)
- » [Manipulando Tablas de datos en IOS con StoryBoards usando UML y TDD](#)

Últimas ofertas de empleo

- 2011-09-08 [Comercial - Ventas - MADRID.](#)
- 2011-09-03 [Comercial - Ventas - VALENCIA.](#)
- 2011-08-19 [Comercial - Compras - ALICANTE.](#)
- 2011-07-12 [Otras Sin catalogar - MADRID.](#)
- 2011-07-06 [Otras Sin catalogar - LUGO.](#)

 Roberto Canales Mora
rcanalesmora

rcanalesmora Tutorial: Control de la calidad, aseguramiento de la calidad y calidad total en el desarrollo de software [adictosaltrabajo.com/tutoriales/tut...#xpd](#) 42 minutes ago · reply · retweet · favorite

rcanalesmora @Mouleb Supongo que encontrarás mi último libro en cualquier librería grande. Editorial [starbook.es/libros/PLANIFI...](#) 20 hours ago · reply · retweet · favorite

rcanalesmora @Mouleb @recena Habitualmente usamos PivotalTracker pero a los clientes solemos recomendar #atlassian pero nos adaptamos #Scrum yesterday · reply · retweet · favorite

 Join the conversation

Control de calidad

Ciclo de construcción

T ● Versiones preliminares → Controles intermedios → Control Final Calidad ●

Aseguramiento calidad

- Usar repositorios
- Programar con TDD
- Medir continuamente
- Retrospectivas

Medidor de calidad

Búscanos en Facebook

Roberto Canales en Facebook

Me gusta

A 102 personas les gusta Roberto Canales en Facebook.

Plug-in social de Facebook

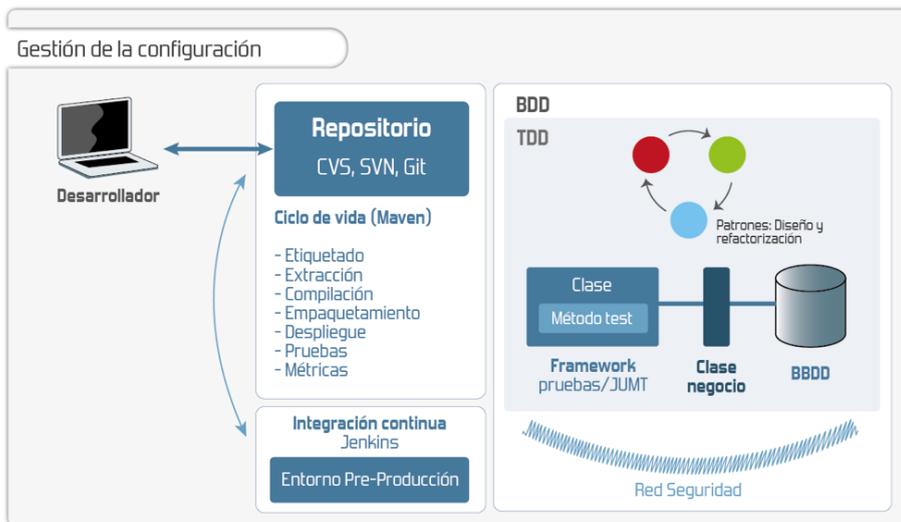
Para mi, las medidas de aseguramiento de la calidad tiene que estar a tres niveles:

1 - Gestión de la configuración y técnicas de XP, entre las que incluimos :

- Uso eficiente de repositorios de código, aunque haya un sólo programador.
- Desarrollo en base a TDD: primero hacer los test y luego los programas para tener garantías de que no rompemos cosas al arreglar otras. Lo podríamos sofisticar con BDD.
- Definición del ciclo de vida del proyecto externo a herramientas particulares:
 - Etiquetado, extracción de código, compilación, empaquetamiento, despliegue, ejecución de pruebas, obtención de métricas de calidad.
- Uso de integración continua: para que todo lo anterior se haga cada vez que se cambia un fuente, sin intervención humana.
- Aplicación de técnicas avanzadas de desarrollo OOP, patrones de diseño y de refactorización.

Es importante destacar que podemos obtener métricas automáticas:

- Porcentaje de líneas duplicadas.
- Complejidad del código.
- Porcentaje de código cubierto por test automáticos.
- Nivel y criticidad de ruptura de reglas generales y particulares.
- Gráfica de evolución de roturas de test automáticos y no compilaciones automáticas.



2 - Metodologías ágiles

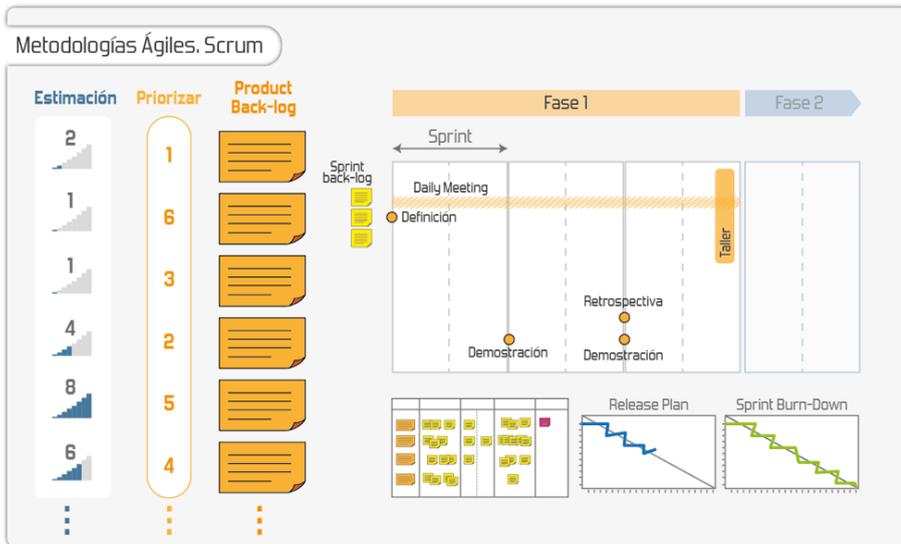
El agilismo está siendo una revolución en la industria porque proporciona una ventaja importante:

- Descomponer un problema grande en unidades manejables.
- Involucrar a negocio desde el principio en la definición y priorización.
- Transmitir la necesidad a los equipos de adquirir una técnica de estimación.
- Trabajar en ciclos cortos con presentación regular de resultados completos a cliente.
- Obtención de realimentación por parte de los usuarios.
- Mantenimiento del pulso del equipo con reuniones diarias.
- Enfoque en la mejora continua con retrospectivas y talleres.

Obtenemos automáticamente métricas de:

- Velocidad de desarrollo del equipo.

- Avance global del proyecto.
- Desviación entre estimaciones y realidad: lo que puede ayudar a aprender a mejorar las técnicas de estimación.



3 - Frameworks y arquitecturas:

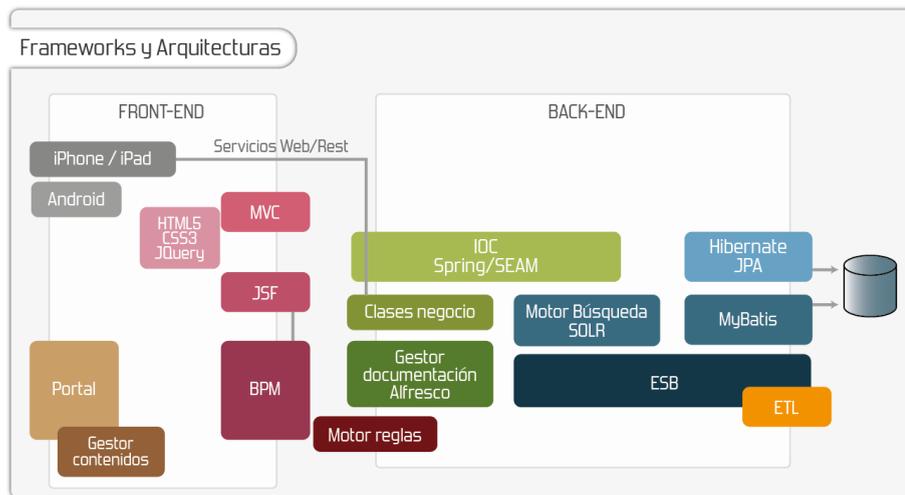
Aunque el desarrollo informático cambia constantemente hay componentes, de más o menos nivel, que se van constituyendo como estándares de facto en la industria porque solucionan de un modo estándar decenas de problemas. Si desarrollamos soluciones a medida tenemos que invertir tiempo en que mejoren. Si usamos elementos utilizados y desarrollados por miles de personas, sin hacer, nada estos evolucionan y se adaptan a nuevas necesidades.

Parece sensato al menos conocerlos para determinar su idoneidad al aplicarlos a un problema específico.

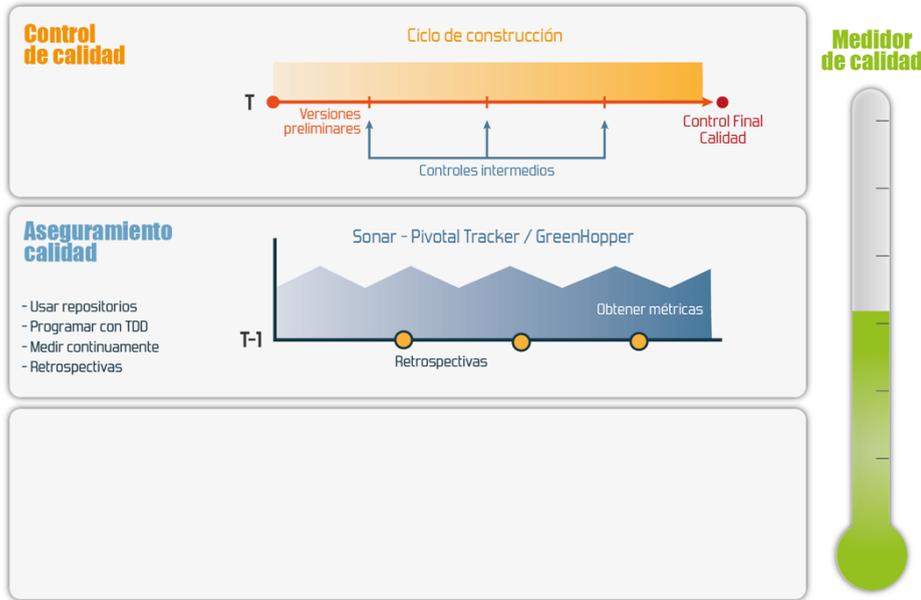
Muchos desarrolladores simplemente los descartan por la falta de control que les provoca no haberlos construido ni poder seguir sus cambios. Esto puede ser un síntoma de inmadurez o un exceso de individualismo. Un grupo de profesionales puede tener un conocimiento general de todos los componentes y cada individuo especializarse en uno de ellos. Con una buena gestión del conocimiento, en el día a día, un equipo puede ser extremadamente eficiente. El conocimiento es demasiado basto como para heroicidades.

Tampoco está mal recordar que muchos buenos profesionales tienen prejuicios contra los frameworks porque mucha gente "apaga su cerebro", sin querer entender su funcionamiento de un modo profundo, creando unos graves problemas en una madeja todavía más gorda que si no se usasen.

He aquí algún ejemplo de piezas estándar que podemos utilizar en proyectos Java.



Cuando empezamos a medir podemos detectar puntos de mejora. Eso sí, las métricas no nos pueden esclavizar porque son un medio y no un fin en si mismo.



4. Mejora continua y calidad total

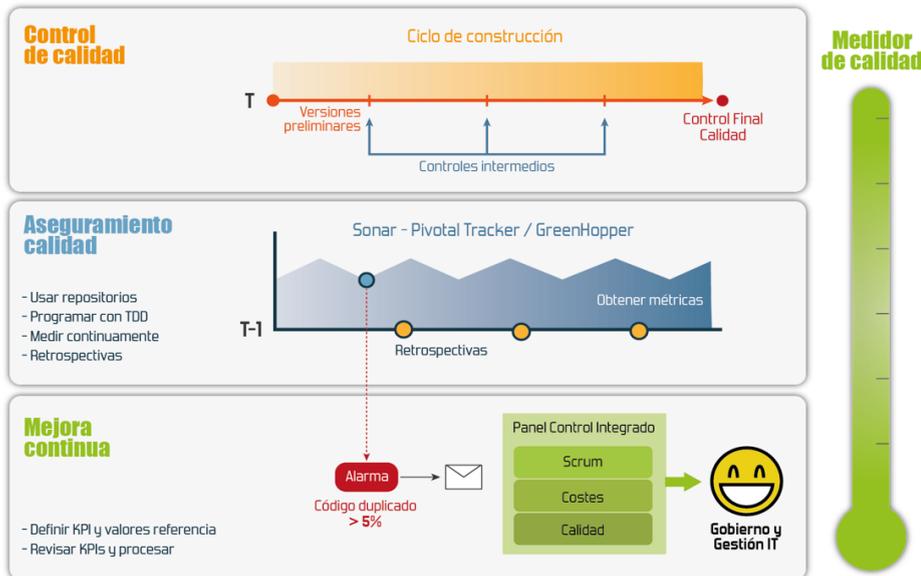
No debemos olvidar que para negocio muchas veces las áreas de tecnología son esos reductos de frikis a los que nadie entiende, que siempre gastan más de lo que se espera, ponen pegajos por hacer más cosas en paralelo, no saben reportar en la forma e inmediatez deseada, se retrasan y rompen cosas al arreglar otras siendo poco fiables. Esto ha sido un gran argumento para descapitalizar las empresas con outsourcing descomunales.

Hay que comprender que no sólo hay que hacer un buen trabajo técnico desde nuestra perspectiva, sino que hay que satisfacer las necesidades, en los aspectos anteriores, de los patrocinadores: de negocio. Sobre todo si queremos conservar nuestros puestos de trabajo ;-)

El responsable de tecnología tiene que gestionar: si no se manejan números no se está gestionando y esto se debe hacer en muchos frentes:

- Manejar un portafolio de proyecto y tareas críticas, y disponer de mecanismos sencillos para comunicar. El agilismo proporciona métricas, y mejor todavía, cercanía con negocio que sabe exactamente cómo está el proyecto porque están dentro de él.
- Tomar medidas correctoras en tiempo temprano: las métricas de calidad automáticas ayudan a captar olores. Las métricas ágiles también.

Los técnicos tienen que ponérselo fácil a los gestores no viendo esta gestión como un enemigo, sino como una imperiosa necesidad. Lo ideal es contar con unos indicadores numéricos clave (KPIs) para ayudar en la gestión: pocos pero útiles. Estas métricas deben integrar situación de proyectos, niveles de calidad y costes.



Cuando alguien me dice que prefiere un lenguaje respecto a otros porque su curva de aprendizaje es mejor cuanto menos le miro con caras raras, porque el lenguaje es sólo una parte, importante, pero una parte.

Que una persona aprenda a resolver problemas con un lenguaje es sencillo: semanas o meses.

Para que un equipo de gente trabaje como un todo, con fiabilidad y solvencia como el que comentamos, hacen falta muchos años en un entorno disciplinado, eso diferencia profesionales. Eso diferencia empresas.

Debemos recordar que los costes de software suelen repartirse en un 20% su construcción inicial y un 80% para el mantenimiento, porcentajes que parecen argumentos suficientes para cuidar su calidad y mantenibilidad.

Si habéis visto la película 300 ... la fuerza está en el equipo disciplinado, entrenado y acostumbrado a luchar juntos. :-)



A continuación puedes evaluarlo:

[Regístrate para evaluarlo](#)

Por favor, vota +1 o compártelo si te pareció interesante

[Share](#) |

[1](#)

Anímate y coméntanos lo que pienses sobre este **TUTORIAL**:

» [Regístrate](#) y accede a esta y otras ventajas «



Esta obra está licenciada bajo licencia [Creative Commons de Reconocimiento-No comercial-Sin obras derivadas 2.5](#)

PUSH THIS

Page Pushers

Community

Help?

3
clicks

2 people brought clicks to this page



powered by [karmacrazy](#)