

INTEGRACIÓN CONTINUA

La integración continua es una práctica de desarrollo de software en la cual los desarrolladores combinan los cambios en el código en un repositorio central de forma periódica, tras lo cual se ejecutan versiones y pruebas automáticas.

OBJETIVOS:

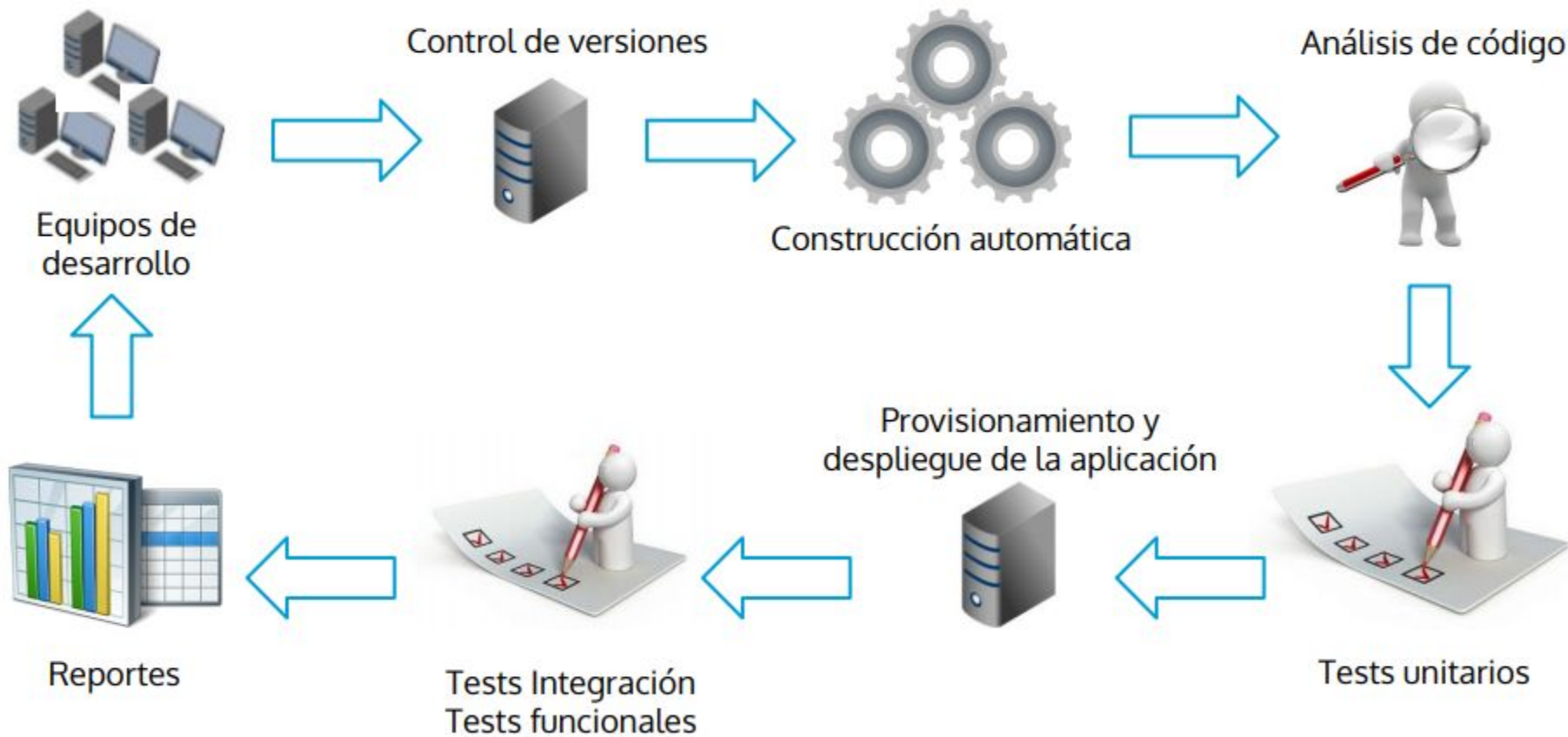
- Encontrar y arreglar errores con mayor rapidez
- Mejorar la calidad del software
- Reducir el tiempo que se tarda en validar y publicar nuevas actualizaciones de software.

¿Por qué es necesaria la integración continua?

Proporcionar las actualizaciones a los clientes con rapidez.

¿En qué consiste la integración continua?

Un servicio de integración continua crea y ejecuta automáticamente pruebas en el código con los nuevos cambios realizados, para identificar inmediatamente cualquier error.



VENTAJAS

- Los desarrolladores pueden detectar y **solucionar problemas de integración de forma continua**, evitando el caos de última hora cuando se acercan las fechas de entrega.
- **Disponibilidad** constante de una versión para pruebas, demos o lanzamientos anticipados.
- **Ejecución inmediata** de las pruebas unitarias.
- **Monitorización continúa** de las métricas de calidad del proyecto.

DESVENTAJA

- Con equipos grandes, el nuevo código se agrega constantemente a la cola de integración, por lo que el **seguimiento de las entregas** (a la vez que se preserva la calidad) **es difícil** y las **compilaciones en cola pueden ralentizar al equipo**.

La integración continua no es necesariamente valiosa si el alcance del proyecto es pequeño o contiene un código heredado no comprobable.

HERRAMIENTAS



Subversion / GIT:

- Herramienta de control de versiones y repositorio de código fuente.
- Sistema centralizado.
- Gestión de históricos de los cambios realizados sobre los ficheros.
- Facilita la ramificación de los proyectos.
- Permite etiquetar los proyectos y su versionamiento.

HERRAMIENTAS



Jenkins

- Identifica las distintas tareas en jobs.
- Mantiene un histórico de la ejecución y el resultado de los mismos.
- Permite lanzar desde compilaciones de aplicaciones java, android, iOS, hasta deploys contra servidores de aplicaciones.
- Notifica a los afectados en caso de eventualidades.

HERRAMIENTAS



TestLink

TestLink

- Afianza la calidad del software
- Permite crear y gestionar casos de prueba
- Organiza los casos de prueba en planes de pruebas
- Gestiona y publica los resultados de los test
- Trazabilidad con los requisitos

HERRAMIENTAS



Sonarqube

- Herramienta utilizada para evaluar la calidad del software.
- Recopila, analiza y permite visualizar las métricas de la calidad del software.
- Realiza análisis estático del código en función de unas reglas configurables y ampliables.
- Nos permite conocer el porcentaje de cumplimiento de las reglas en los proyectos.
- Monitoriza la evolución de la calidad.