

## INTRODUCCIÓN

En el contexto actual, las empresas centran sus esfuerzos en alcanzar la máxima calidad en el software desarrollado. Para ello, se emplean nuevos métodos y técnicas que garanticen la calidad y eviten riesgos o costos.

Con el fin de mejorar la calidad final de los productos, reducir costos, tiempo y aumentar la eficiencia, surge la Integración Continua (IC), uno de los pilares fundamentales sobre los que se sustenta el desarrollo de software.

## PRINCIPIOS ÁGILES RELACIONADOS



Nuestra mayor prioridad es **satisfacer al cliente**

Entregar software funcional frecuentemente



## MEJORES PRÁCTICAS

- Mantener un repositorio de fuente única
- Automatizar la construcción
- Realizar una autocomprobación de la construcción
- Cada compromiso debe basarse en una máquina de integración
- Mantener la construcción rápida
- Prueba en un clon del entorno de producción
- Facilitar a cualquiera la obtención de la última versión ejecutable
- Todos pueden ver lo que está pasando
- Automatizar la implementación



**VIDEO PARA AYUDARTE**



**UTN - FACULTAD REGIONAL CÓRDOBA  
INGENIERÍA EN SISTEMAS DE INFORMACIÓN  
INGENIERÍA DE SOFTWARE - 4K1 - 2020 - GRUPO 10**

Casella, Juan Manuel  
Coronel, Ayelén del Milagro  
Farfan, Fabio Hugo

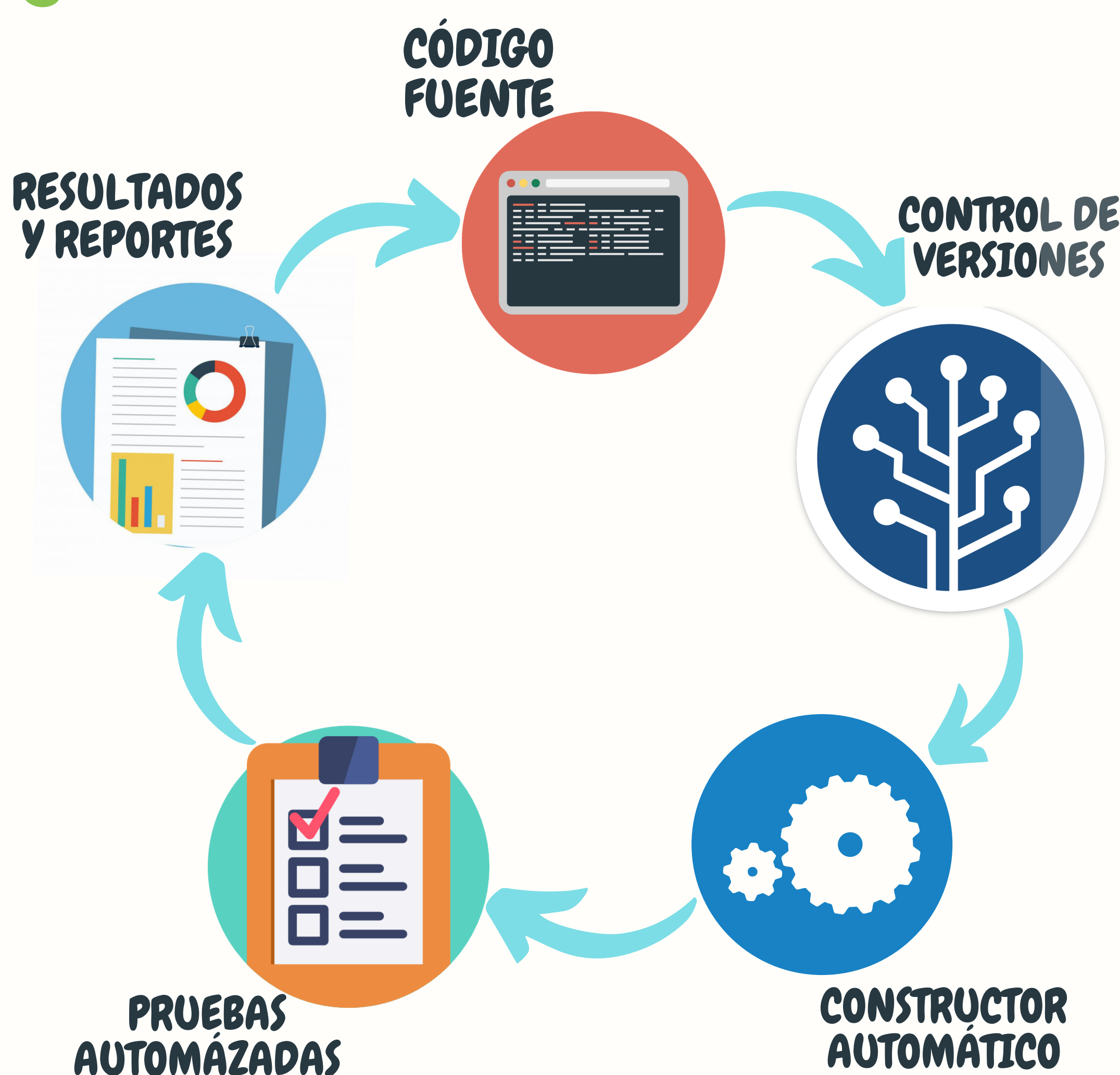
Ledesma, Juan Ignacio  
Mellibosky, Joaquín  
Rudy, Franco Ignacio



## ¿QUÉ ES LA IC?

La integración continua es una **práctica de desarrollo de software** en la que los miembros de un equipo integran su trabajo con frecuencia (por lo menos una vez al día), lo que lleva a múltiples integraciones. Cada integración se verifica mediante una compilación automatizada (incluida la prueba) para detectar errores de integración lo más rápido posible.

## ¿CÓMO FUNCIONA?



Los desarrolladores revisan el código en sus espacios de trabajo, cuando terminan, confirman los cambios en el repositorio.

El servidor de IC monitorea el repositorio, verifica los cambios cuando ocurren, construye el sistema y ejecuta pruebas unitarias y de integración. Luego libera artefactos desplegables para probar. El servidor IC asigna una etiqueta de compilación a la versión de código que acaba de compilar e informa al equipo de la compilación exitosa. En caso de que la compilación o las pruebas fallen, el servidor alerta al equipo para que lo solucione lo antes posible.

Se continúa integrando y probando continuamente durante todo el proyecto.

## VENTAJAS



Detectar **errores de compilación** rápidamente



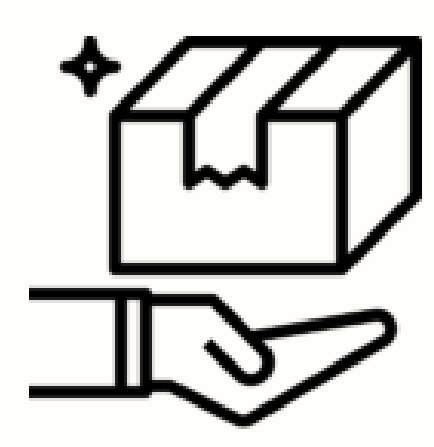
Detectar **funcionamientos anómalos** en el software



Mejorar la **calidad**

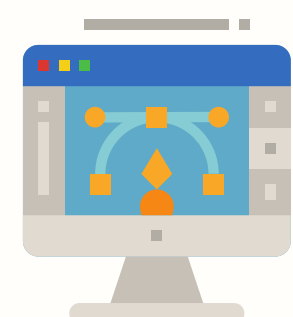


Mejorar la **comunicación**



Entregas con mayor **rapidez y frecuencia**

## CONCLUSIÓN



La IC es una práctica valiosa y bien establecida en las organizaciones modernas de ingeniería de software de alto rendimiento debido a que permite incrementar la eficacia y la eficiencia de los resultados del equipo de desarrolladores.

Es una práctica recomendada de metodología ágil, ya que permite a los equipos de desarrollo de software centrarse en cumplir los requisitos comerciales, la calidad del código y la seguridad porque los pasos de implementación están automatizados.

## REFERENCIAS

<https://martinfowler.com/articles/continuousIntegration.html>

<https://www.atlassian.com/continuous-delivery/continuous-integration>

<https://www.campusmvp.es/recursos/post/integracion-continua-que-es-y-por-que-deberias-aprender-a-utilizarla-cuanto-antes.aspx>

