

*Reducción de costos de desarrollo
asegurando la calidad del
software.*

Agenda.

Nuestra Percepción de la calidad

**Testing como parte del
aseguramiento de la calidad**

**Ubicación de los defectos dentro
de las fases de desarrollo**

Relación “COSTO-CALIDAD”

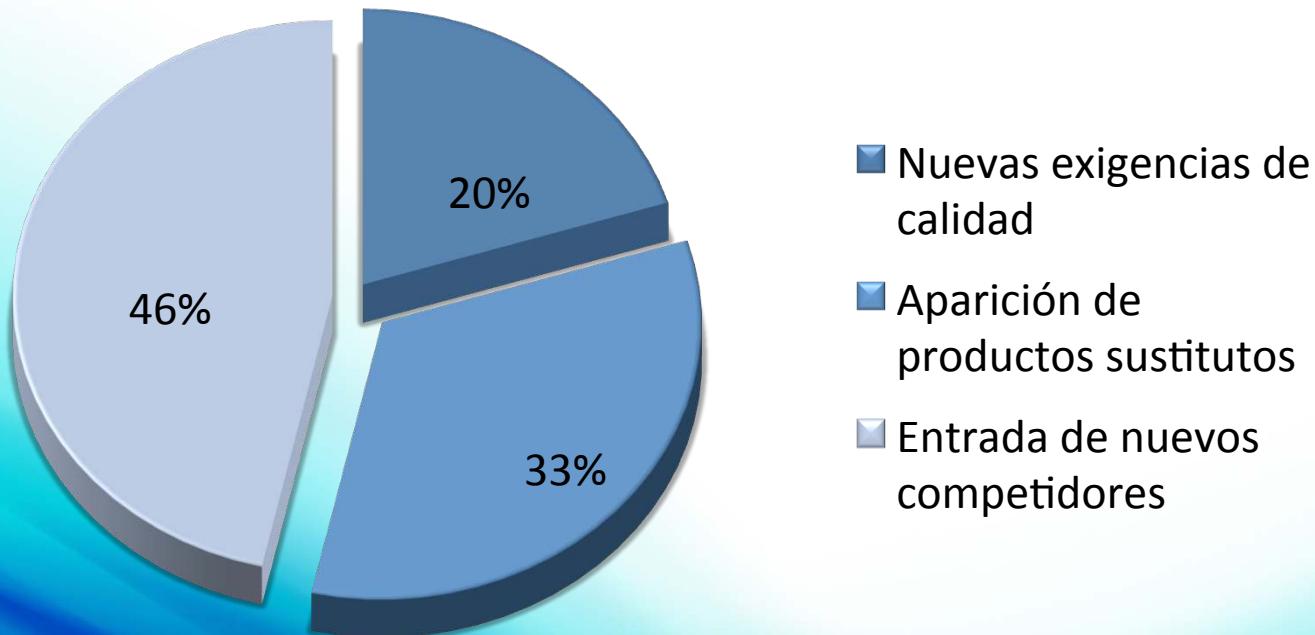
Agenda (cont).

El valor de testing

El ROI de testing

Conclusiones

Causas del incremento de la competitividad... (y la calidad).



Fuente: Instituto de Dirección y de Organización de Empresas (junio 2011)

Percepción de la calidad.... (y su costo)



=
=



Integración del equipo de QA



- Procesos.
- Compromiso de la gerencia.
- Experiencia del personal.
- Productos finales.
- Uso de herramienta.
- Métricas.
- Ambiente para testing.
- Datos de prueba.
- Administración de cambios.
- Concientización de desarrolladores.



Sacrificio de la calidad



Calidad del software



**SOFTWARE
TESTING**



¿Por qué las pruebas son necesarias?

Los errores causan defectos que pueden provocar, mala imagen de nuestro equipo ante nuestros clientes, retrabajo innecesario, una “tortuosa” implantación del nuevo desarrollo y sus nuevas versiones, sin olvidar el alto costo de realizar correcciones en las últimas fases del proyecto.



Ciclo de vida del software

Definición

Diseño

Construcción

Validación

Entrega

Actividades de pruebas

Administración de defectos

Pruebas estáticas

Planeación de pruebas

Preparación de pruebas

Pruebas
unitarias

Pruebas de integración

Pruebas de sistema

Pruebas de
aceptación

¿Por qué las pruebas son necesarias?



Si realizamos pruebas tanto en la documentación como en el software, reducimos el **RIESGO** de encontrar **PROBLEMAS** en un ambiente productivo y podemos ubicar áreas de oportunidad dentro de nuestros desarrollos.

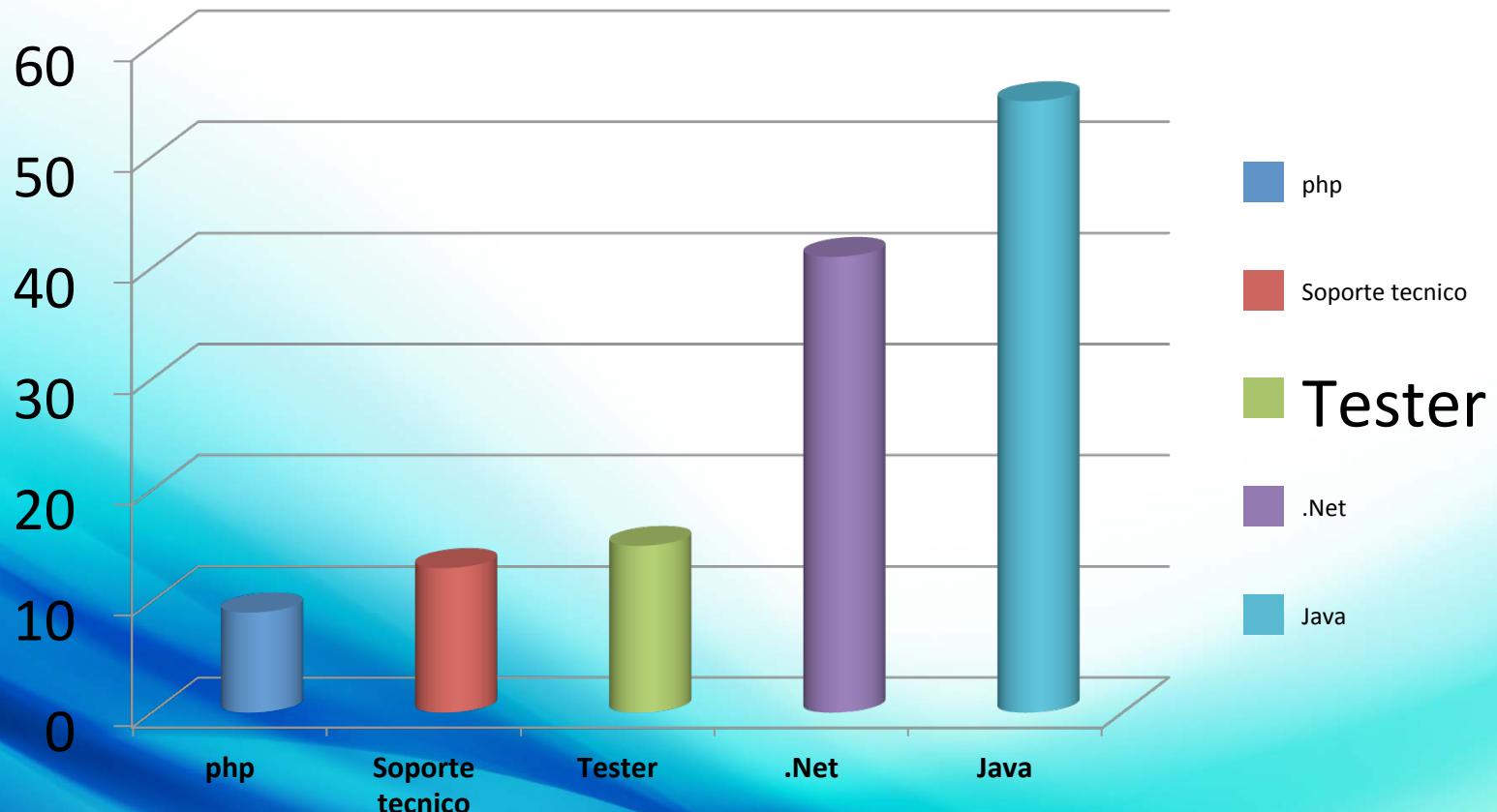
Beneficios de las pruebas

Las pruebas contribuyen a la calidad del software y cumplir con los estándares requeridos.



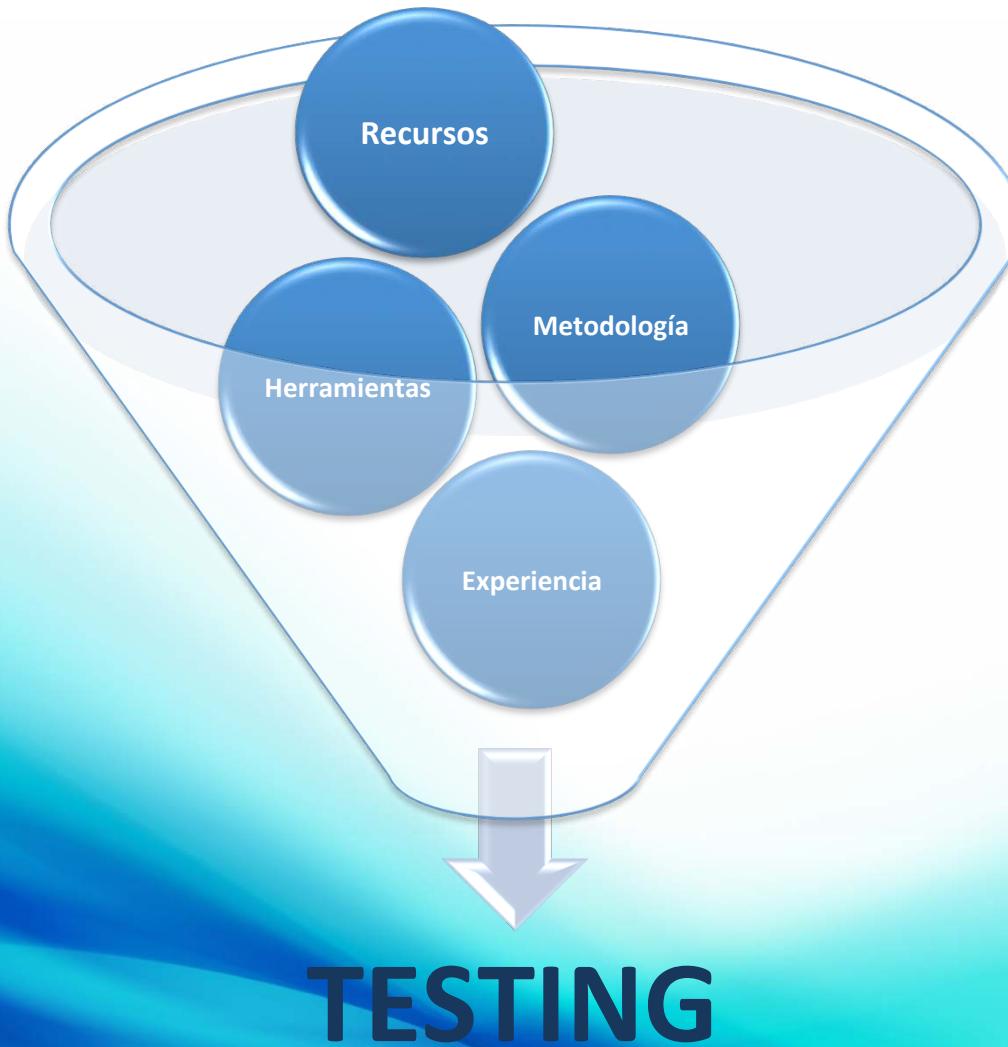
Testing dentro del aseguramiento de la calidad.

Número de vacantes solicitadas por especialidad



Fuente: <http://empleosti.com.mx/blog/top-5-de-tecnologias-mas-solicitadas-por-el-mercado-laboral-2012/>

Características para tener éxito en la implantación de testing.



TESTING



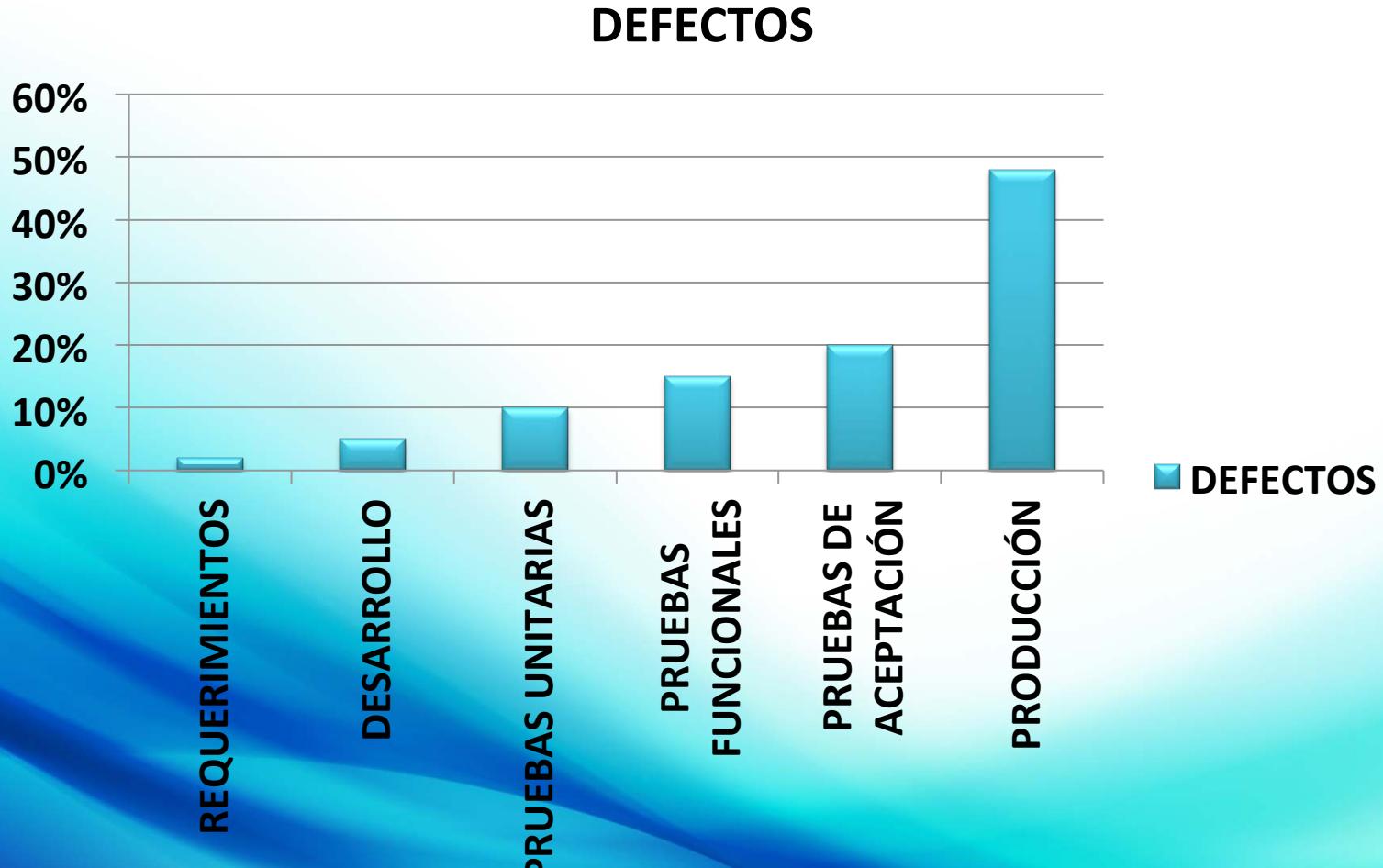
Pruebas de Performance
y stress

Pruebas Funcionales

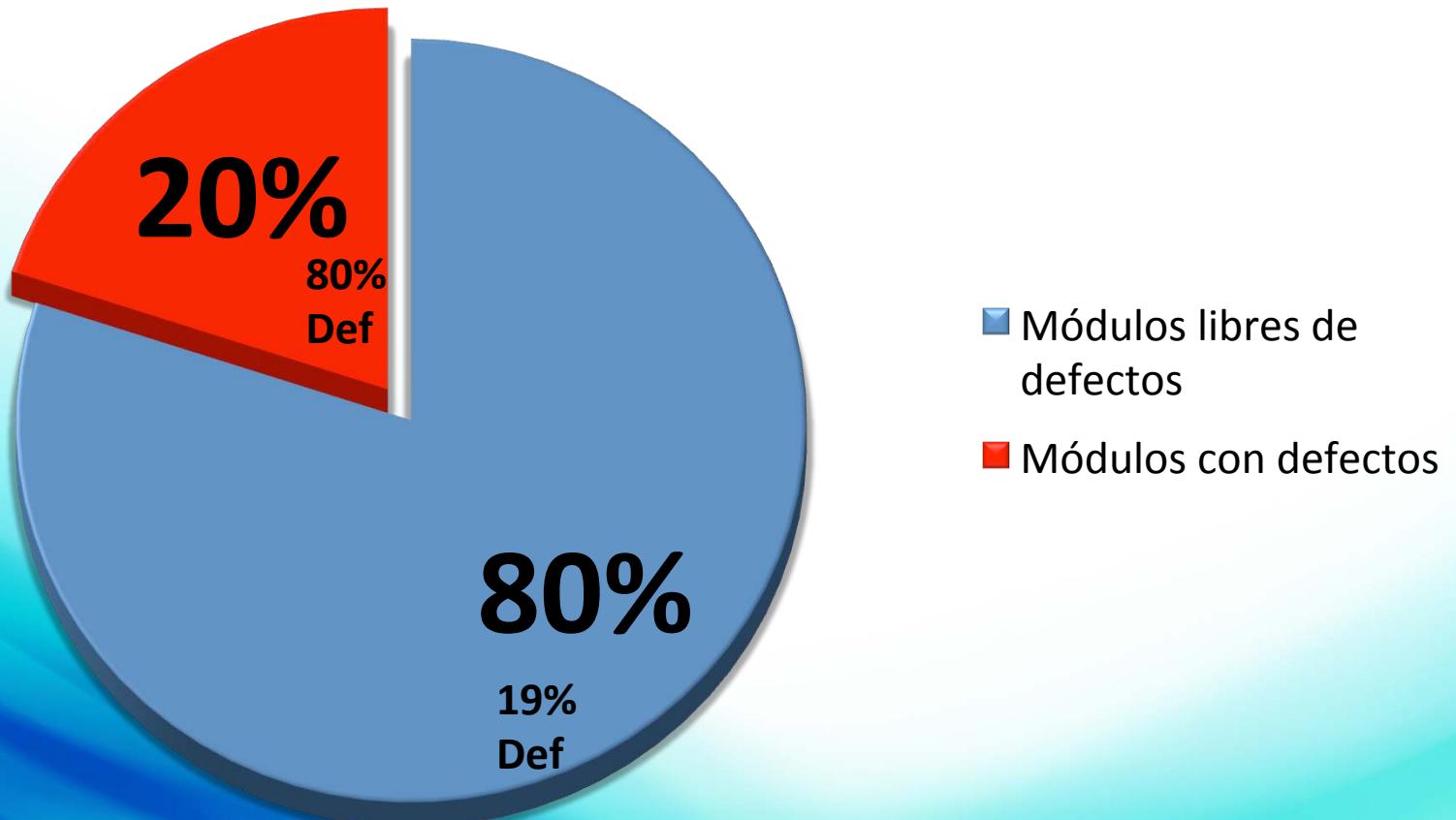
Automatización de
pruebas

Análisis y Monitorización
en producción

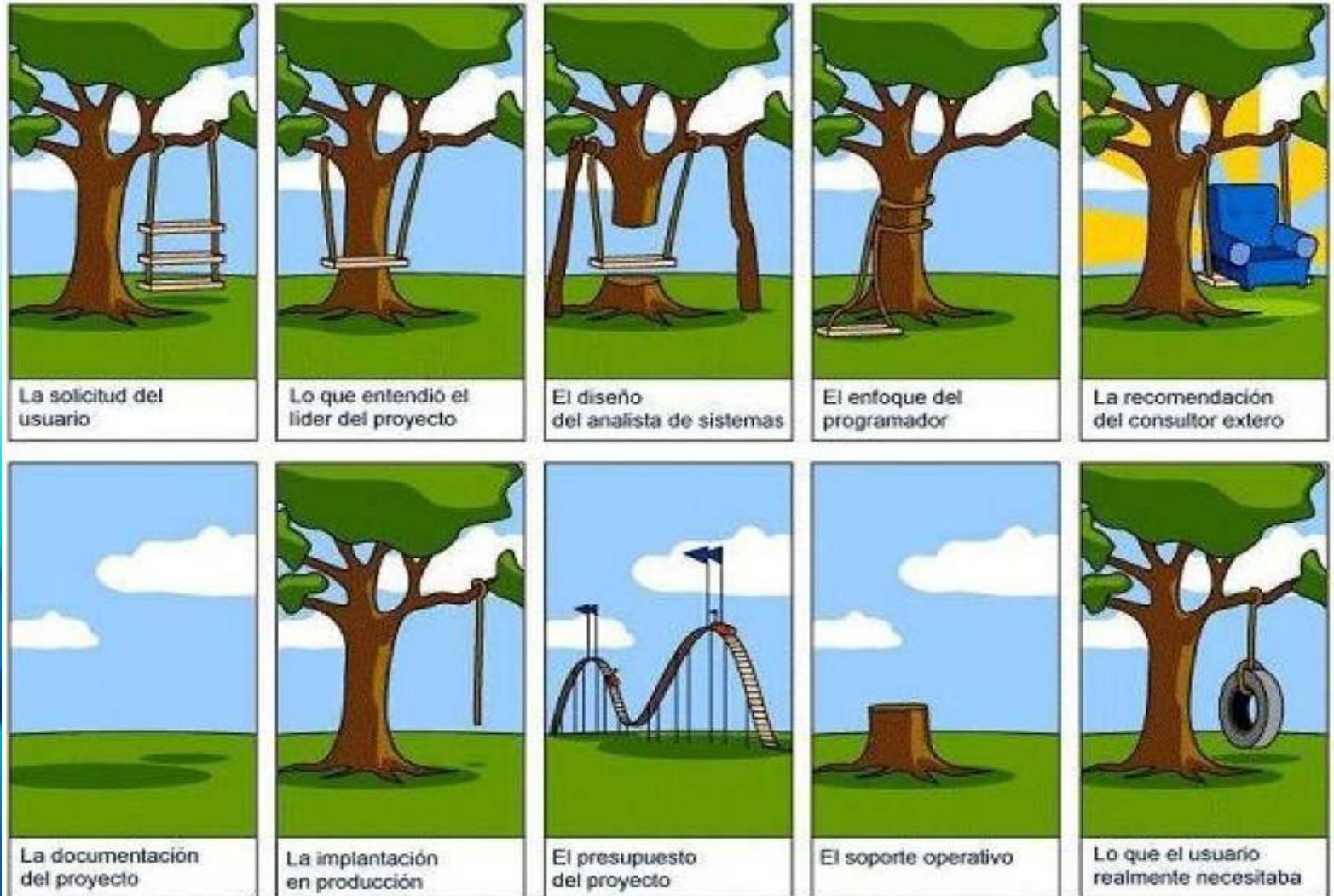
Ubicación de defectos por fase de desarrollo



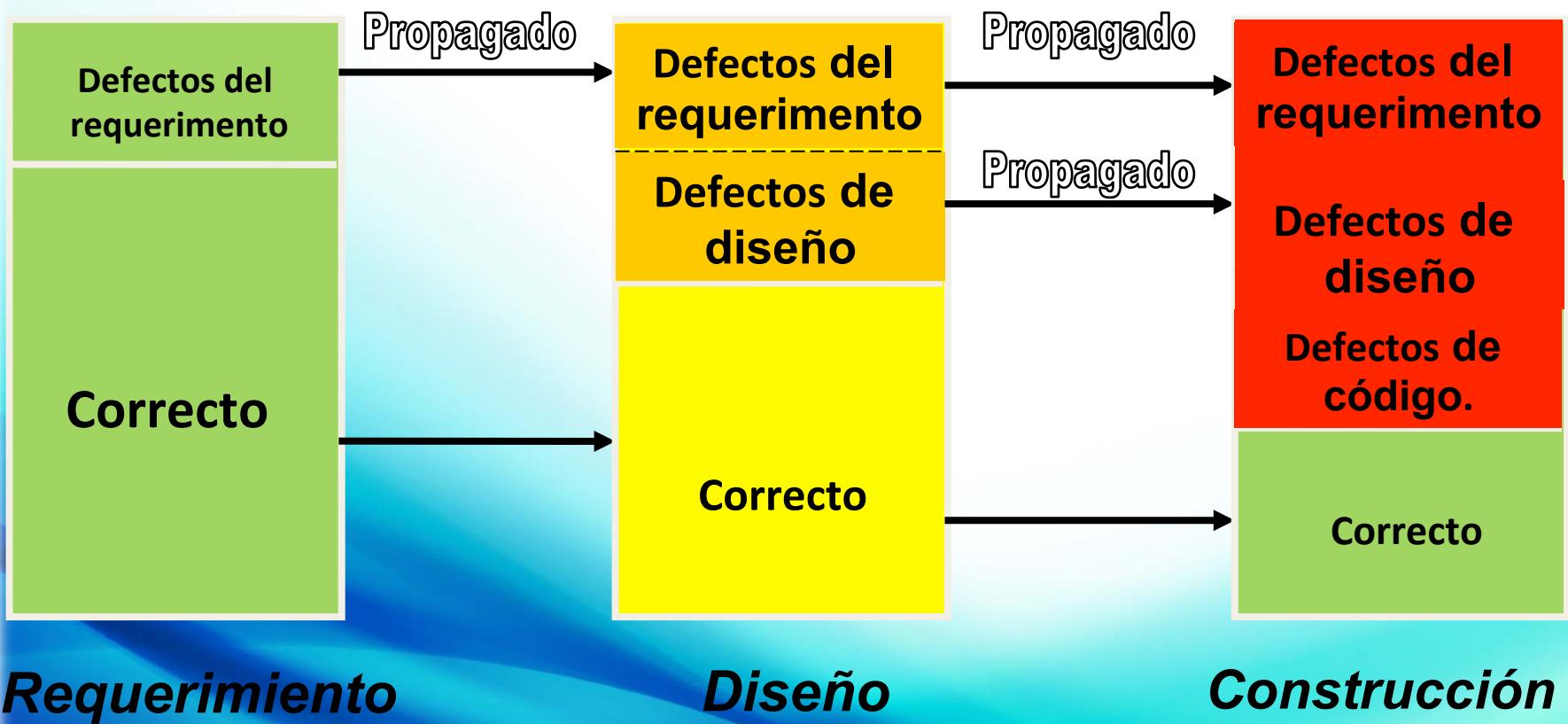
Ubicación de los defectos



“Ciclo de vida del software” (La definición del proyecto y el teléfono descompuesto)

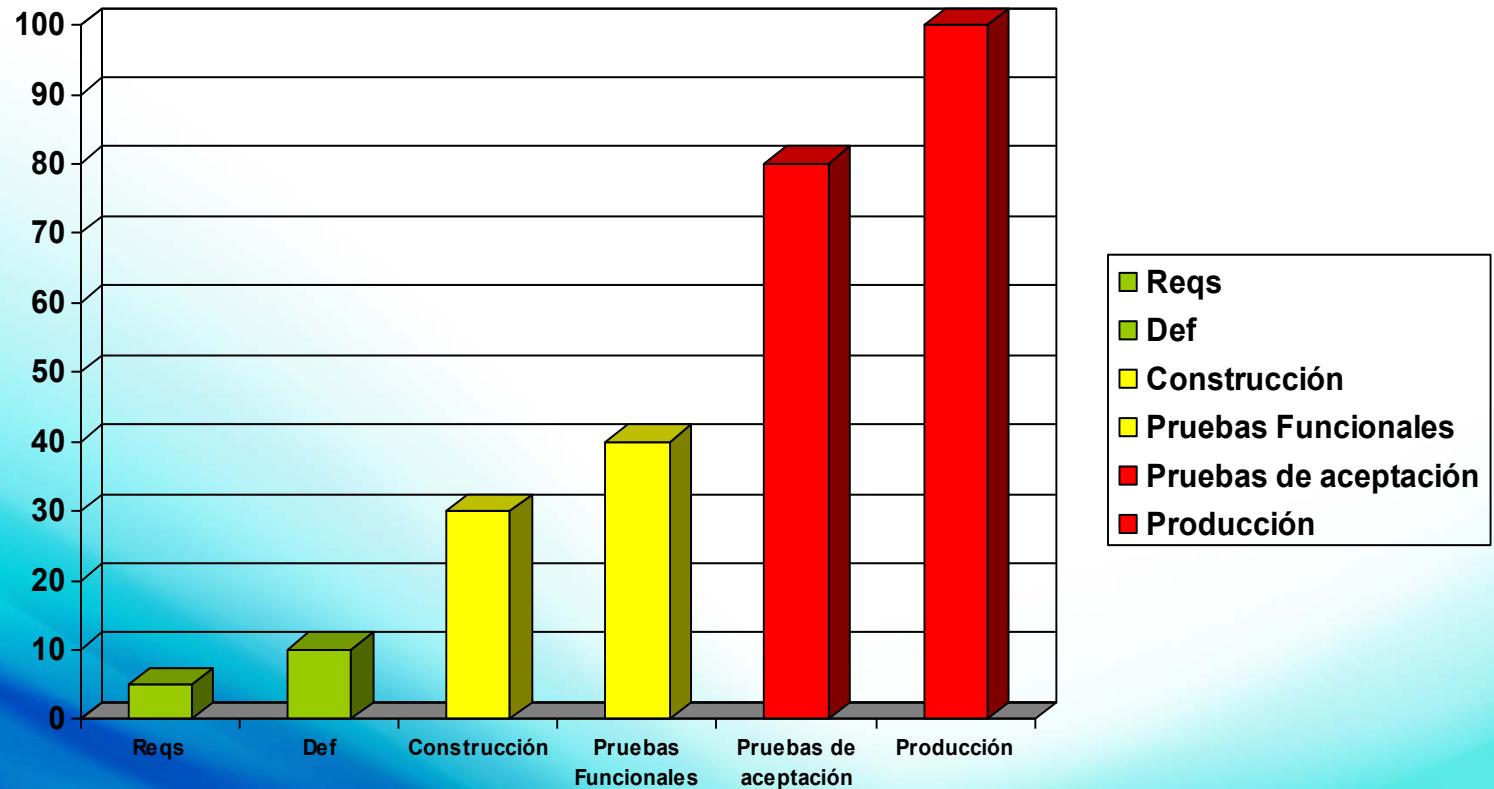


La propagación de defectos en contra de la calidad

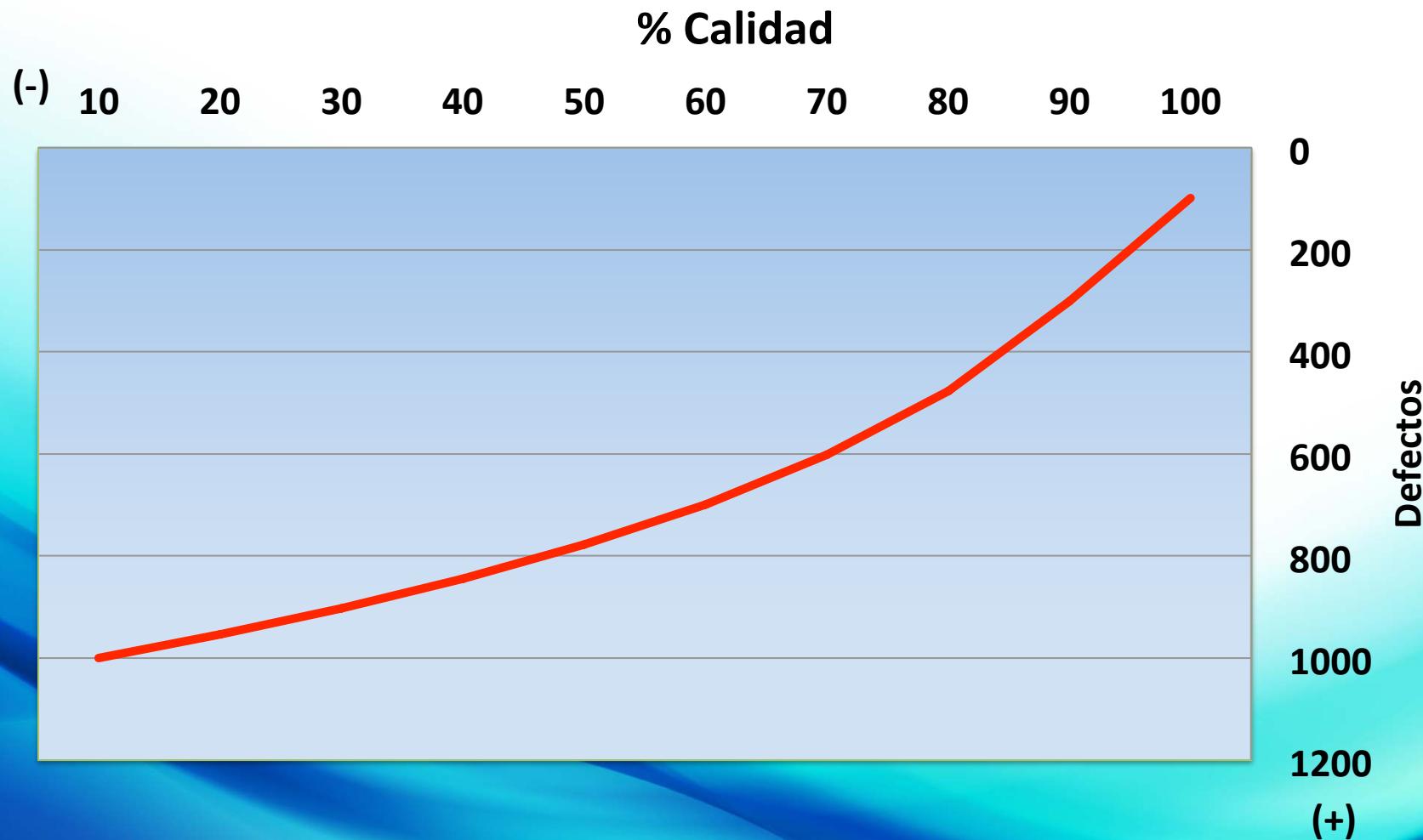


Relación Costo corrección defectos -vs-Fase de desarrollo

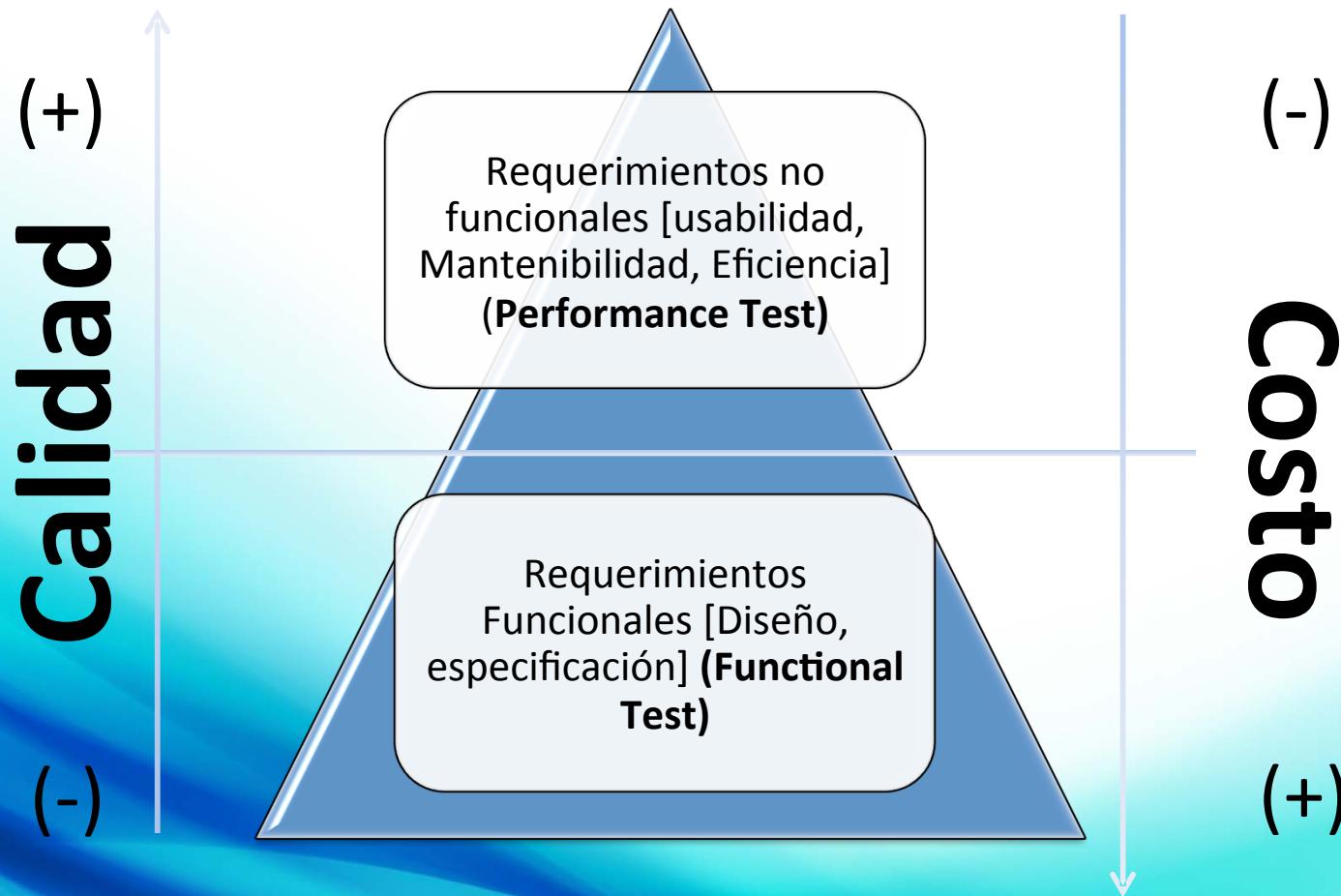
(+)



Relación Defectos -vs- Calidad



Relación Costo-Calidad



Beneficios de las pruebas.

Foco en las actividades

- El equipo de pruebas coadyuva a que el equipo de desarrollo sólo se dedique a programar.

Mayor cobertura de pruebas

- La especialización del equipo de pruebas permite ver más casos de prueba y con esto encontrar un número mayor de defectos.

Beneficios de las pruebas.

Metodología

- La metodología de pruebas promueve que se documenten los procesos de desarrollo permitiendo que estos sean auditables y verificables.

Aseguramiento de la Calidad

- Con la metodología del equipo de pruebas se asegura que el software trabaja de manera óptima según se definió.

Beneficios de las pruebas.

Confiabilidad

- El cliente tiene la confianza de que el equipo de desarrollo asegurará el éxito del proyecto para ambas partes.

Comunicación

- El equipo de pruebas promueve lazos de comunicación entre el equipo de desarrollo y el usuario, al tender un puente de mejor entendimiento.

Beneficios de las pruebas.

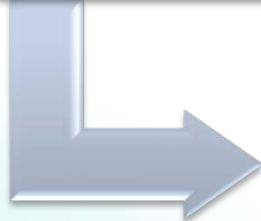
Costos

- Al encontrar defectos de manera temprana se reducen los costos del proyecto.

Garantía de servicio.

- También la metodología del equipo de pruebas permite una adecuada identificación de defectos –vs- nuevos requerimientos relacionados con el desarrollo, una vez que éste se encuentra en producción.

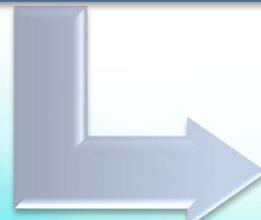
Satisfacción
al cliente



Garantía de
en la calidad
del software



Reducción
de COSTOS

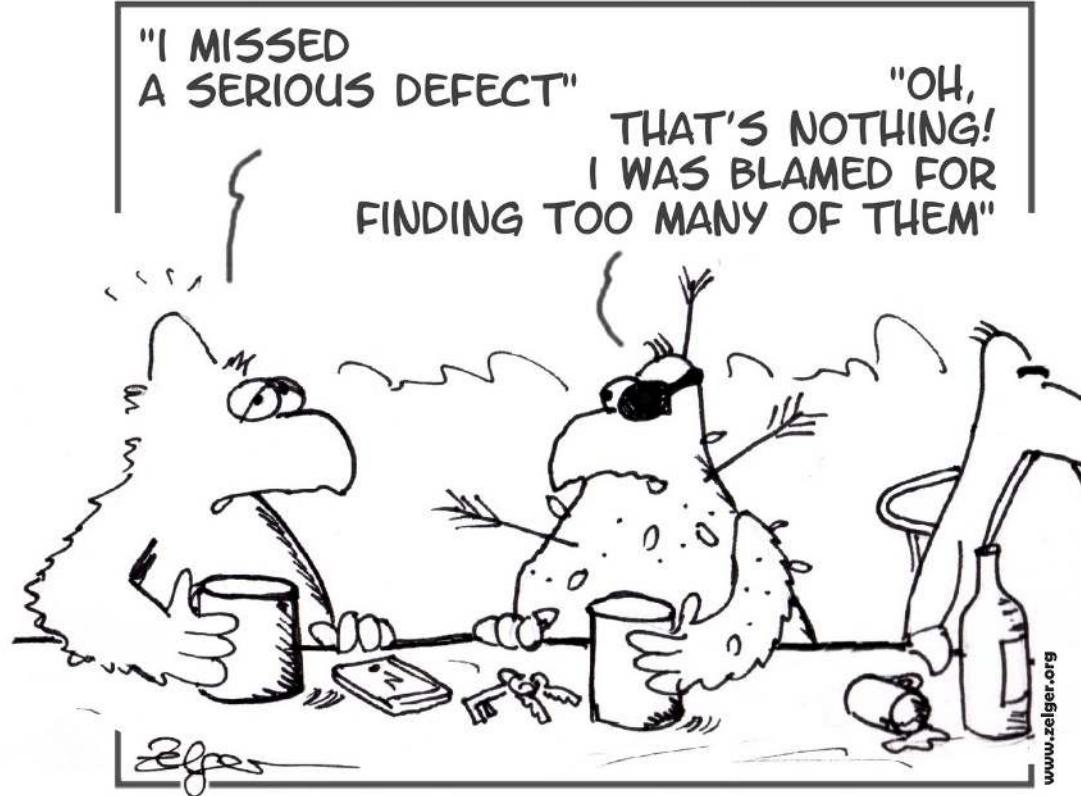


TESTING

Beneficios de las pruebas.

“La calidad no es intangible.
El propósito de “TESTING” es hacer que la calidad sea visible.
¡TESTING es la medida de la calidad del software!”.

Bill Hetzel 1988, Autor del libro: “The Complete Guide to Software Testing”



El valor de testing (“TESTING VALUE”)

$$TV = \frac{1}{2} [\sum DCDP * CCED] + \frac{1}{2} [\sum (DCDP * DCCP)] + SBLR - TBL$$

- TV (Testing Value)
- DCDP (Defect Corrections per Development project Phase)
- CCED (Project Support Correction Cost per End-user Defect)
- DCCP (Average Development Correction Cost by project Phase)
- SBLR (Software Business Loss due to testing Risks this project's business sponsor assumed in lieu of correction and, in fact, the risk occurred).
- TBL (Testing Business Loss due to incomplete test planning or test execution that allowed business loss to occur).

LA EXPRESIÓN [DCDP * SCCED]

Representa el valor generado por el equipo de pruebas con respecto al equipo de desarrollo al reducir el numero de defectos injectados al ambiente productivo.

LA EXPRESIÓN [DCDP * DCCP]

Representa el valor generado por el equipo de pruebas con respecto al equipo de soporte al reducir el numero de defectos injectados al ambiente productivo y detectados por el usuario final.

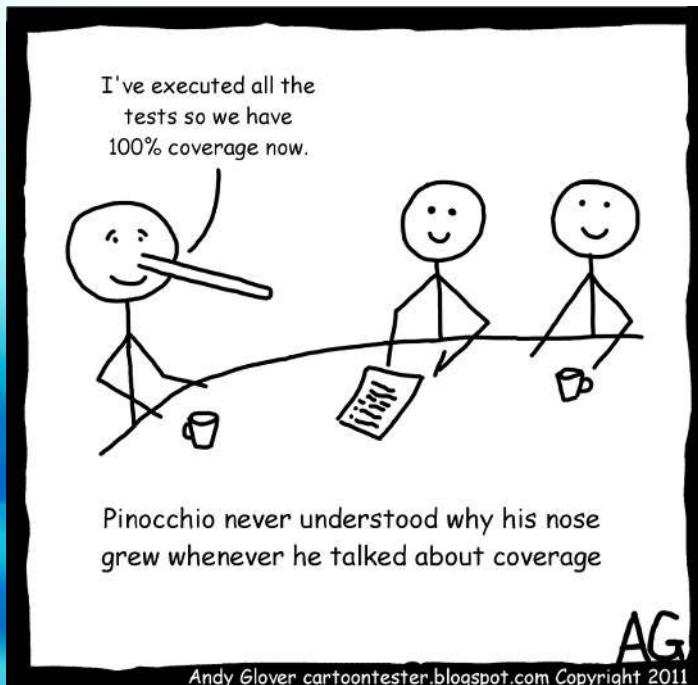
EL FACTOR [SBLR]

Representa las consecuencias de que el patrocinador del proyecto asuma un riesgo reportado por el equipo de testing y no se hayan tomado las medidas necesarias para mitigarlo e inclusive evitarlo.



EL FACTOR [TBL]

Representa las consecuencias de que el equipo de pruebas no haya terminado la ejecución de las pruebas planeadas, que haya omitido la ejecución de algunos casos de prueba o no haya levantado los riesgos inherentes a pruebas y que estos se hayan concretado



El valor de testing (“TESTING VALUE”)

$$TV = 1/2 [\sum DCDP * CCED] + 1/2 [\sum (DCDP * DCCP)] + SBLR - TBL$$

Here are the Basili and Boehm's cost factors per development phase.

- Design & Code 461 defects @ \$139/correction
- Compile & Bind 2,191 defects @ \$455/correction
- Pre-Prod Integration 317 defects @ \$977/correction
- Post Release 25 defects @ \$14,102/correction

$$\frac{1}{2} [\sum DCDP * CCED] \quad 1/2 (461+2,191+317) * \$14,102 = \$20.9M$$

$$\frac{1}{2} [\sum (DCDP * DCCP)] \quad 1/2 (461 * \$139 + 2,191 * \$455 + 317 * \$977) = \$0.7M$$

SBLR: \$4M

TBL: \$2M

$$TV = \$20.9M + \$0.7M + \$4M - \$2M = \$23.6M$$

Testing ROI (Retorno de la inversión de testing)

Testing Investment Options: ROI Analysis				
	No Formal Testing	Manual Testing	Automated Testing	
Testing Staff	\$ -	\$ 60,000.00	\$ 60,000.00	\$ 60,000.00
Infraestructure	\$ -	\$ 10,000.00	\$ 10,000.00	\$ 10,000.00
Tools	\$ -	\$ -	\$ 12,500.00	\$ 12,500.00
Total Investment	\$ -	\$ 70,000.00	\$ 82,500.00	\$ 82,500.00
Development				
Must-Fix Bugs Found	\$ 250.00	\$ 250.00	\$ 250.00	\$ 250.00
FixCost (Internal Failure)	\$ 2,500.00	\$ 2,500.00	\$ 2,500.00	\$ 2,500.00
Testing				
Must-Fix Bugs Found	\$ -	\$ 350.00	\$ 500.00	\$ 500.00
FixCost (Internal Failure)	\$ -	\$ 35,000.00	\$ 50,000.00	\$ 50,000.00
Customer Support				
Must-Fix Bugs Reported	\$ 750.00	\$ 400.00	\$ 250.00	\$ 250.00
FixCost (External Failure)	\$ 750,000.00	\$ 400,000.00	\$ 250,000.00	\$ 250,000.00
Cost of Quality				
Conformance	\$ -	\$ 70,000.00	\$ 82,500.00	\$ 82,500.00
Nonconformance	\$ 752,500.00	\$ 437,500.00	\$ 302,500.00	\$ 302,500.00
Total CoQ	\$ 752,500.00	\$ 507,500.00	\$ 385,000.00	\$ 385,000.00
Return of Investment	N/A	725%	467%	

Los TOP TEN para la reducción de defectos (y costos en el QA Assurance)

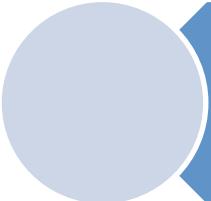
1. Encontrar y corregir un problema de software después de su liberación a producción, es 100 veces más caro que encontrarlo y corregirlo durante la fase de requerimiento.
2. Los proyectos actuales de software gastan cerca del 40 al 50% del esfuerzo en re-trabajo que se pudo evitar.
3. Cerca del 80 % del re-trabajo proviene de un 20% de defectos.
4. Cerca del 80 % de los defectos provienen del 20 % de los módulos iniciales, pero cerca de la mitad de los módulos posteriores se encuentran libres de defectos.
5. Cerca del 90 % del tiempo “muerto” proviene del 10 % de defectos.
6. Las revisiones de los productos de trabajo o las “revisiones entre colegas”, detectan cerca del 60 % de los defectos.
7. Las revisiones dirigidas muestran 35 % más defectos que las revisiones no dirigidas.
8. Implementar mejores prácticas de trabajo puede reducir la tasa de inyección de defectos hasta en un 75 %.

Los TOP TEN para la reducción de defectos (y costos en el QA Assurance)

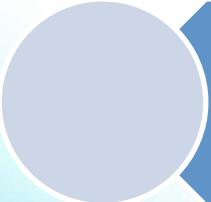
9. El desarrollo de productos de software con alta dependencia cuestan 50 % más por línea de código fuente que los desarrollos de baja dependencia. Sin embargo, la inversión de testing bien vale la pena si el desarrollo involucra significativas operaciones y costos de mantenimiento.
10. Cerca del 40 al 50 % de los “users programs” contienen defectos que no son triviales.

“Software Defect Reduction Top 10 List” by Barry Boehm, University of Southern California and Victor R. Basili, University of Maryland

Conclusión



Un buen aseguramiento de la calidad ayuda en mucho al éxito del proyecto .



Es mejor prevenir que lamentar.



El valor de testing es intangible pero de alto valor económico.



Si no es posible implementar un área de pruebas apóyate de los expertos.



Esto debe de ser un punto de inicio para cambiar la percepción de Testing de ser un “mal necesario” a ser un aliado de alto valor agregado”.

Abhijeet Dixit.

¿Preguntas?

Gracias

federico.kreusch@soluciones-si.com

federico.kreusch@gmai.com

[@FedericoKreusch](https://www.linkedin.com/in/FedericoKreusch)