

## Efectividad de cada revisión:

$$\text{Efectividad} = \left( \frac{\text{Defectos detectados}}{\text{Defectos presentes}} \right) \times 100$$

Diseño:  $\frac{36}{45} \times 100 = 80\%$

Código:  $\frac{63}{87} \times 100 = 74,7\%$

DP: 45 DD: 36 DND: 9

DI: 78 DH: 9 DD: 63 DND: 22

- 8 de cada 10 defectos detectados
- Excelente desempeño
- Poco retrabajo en codificación
- Menor costo de corrección al detectar fallos tempranamente

- 3 de 4 defectos encontrados
- Buen desempeño
- 25% de defectos aún requieren atención
- Posible aumento de carga para testing
- Mayor defectos en producción

## Implicaciones para el proyecto

### Fortalezas:

- Revisión de diseño robusta
- Detección temprana efectiva
- Proceso balanceado

### Áreas de preocupación:

- Brecho de 5.3% entre efectividades
- 22 pasando a pruebas
- Riesgo acumulativo de defectos en producción.

holdas  
oos  
sación  
requisitos  
formados

METRICA	Rev. Diseño	Rev. Código	Diff
Efectividad	80%	74.7%	-5.3%
Defectos detectados	36/45	63/87	
Defectos encontrados	9	22	+13 defectos
Evaluación	Excelente	Aceptable	

## Mejoras Propuestas

#1. Revisiones de código optimizadas (ALTA)  
Obj: Efectividad 74% → 82%

- Acciones:
- Pair Programming
  - Checklist específico
  - Herramientas automatizadas

Impacto esperado:

- Reducir defectos (-32%)
- Ahorro estimado (\$35,000)
- ROI 700%

#2. Estandarización de Procesos (MEDIA)  
Obj: Efectividad 81% entre Fases

- Acciones:
- Documentar Best Practices
  - Replicar técnicas exitosas
  - Entrenar revisiones

Impacto esperado:

- Consistencia
- Reducción de variabilidad
- Cultura de calidad



# Explicación del principio de Pareto

① Un 80% de los efectos provienen del 20% de las causas.

En nuestro caso se ordenó en que 3 de las 8 causas de defectos (fallas) representan el 62.35% de los fallos totales del sistema.

Estas causas son:

DE (234)	
INT (156)	
REQ (125)	
<hr/>	
SIS defectos	
de 826	

② Causas:

#1. Errores de diseño (DES  $\equiv 28.33\%$ )  
1 de cada 4 defectos  $\leftarrow$

#2. Problemas de integración (INT  $\equiv 18.89\%$ )

#3. Requisitos ambiguos (REQ  $\equiv 15.13\%$ )

Oscar Haidar Castrillón

Juan David Díaz

Calidad de Software  
24-07-25

③ Acciones preventivas y correctivas

DES:

- Revisión formal de diseño
- Inspecciones técnicas
- Capacitación técnica
- Herramientas de modelado y validación

INT:

- Integración continua
- Contratos de APIs claros
- Testing de integración robusto
- Doc. técnica
- Protocolo de versionado

REQ:

- Revalidar con stakeholders
- Implementar user-stories
- Sesiones de clarificación
- Capacitación analizando requisitos
- Prototipar
- Establecer revisiones formales

