



INGENIERÍA DE SOFTWARE (IS) MODELO DE PROCESOS DE DESARROLLO

Gabriel Pedraza Ferreira

AGENDA

- Actividades de desarrollo Software
- Modelos de desarrollo
- Documentación
- Errores Comunes



INGENIERÍA DE SOFTWARE

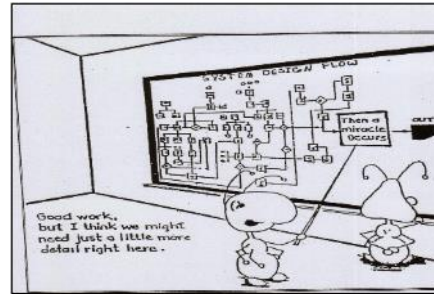
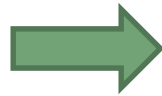
- Desarrollo efectivo de software dentro de los costos y cronograma
- Basada en un enfoque sistemático que usa técnicas y herramientas apropiadas.
- Sigue un proceso predeterminado



ACTIVIDADES DE SOFTWARE



Ingeniería de
requerimientos



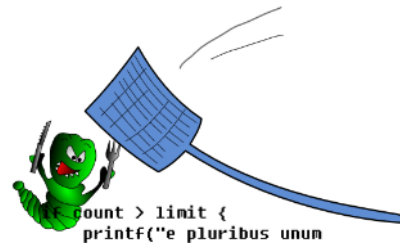
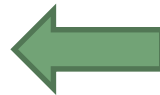
Diseño



Implementación



Mantenimiento



Validación y
Verificación



INGENIERÍA DE REQUERIMIENTOS



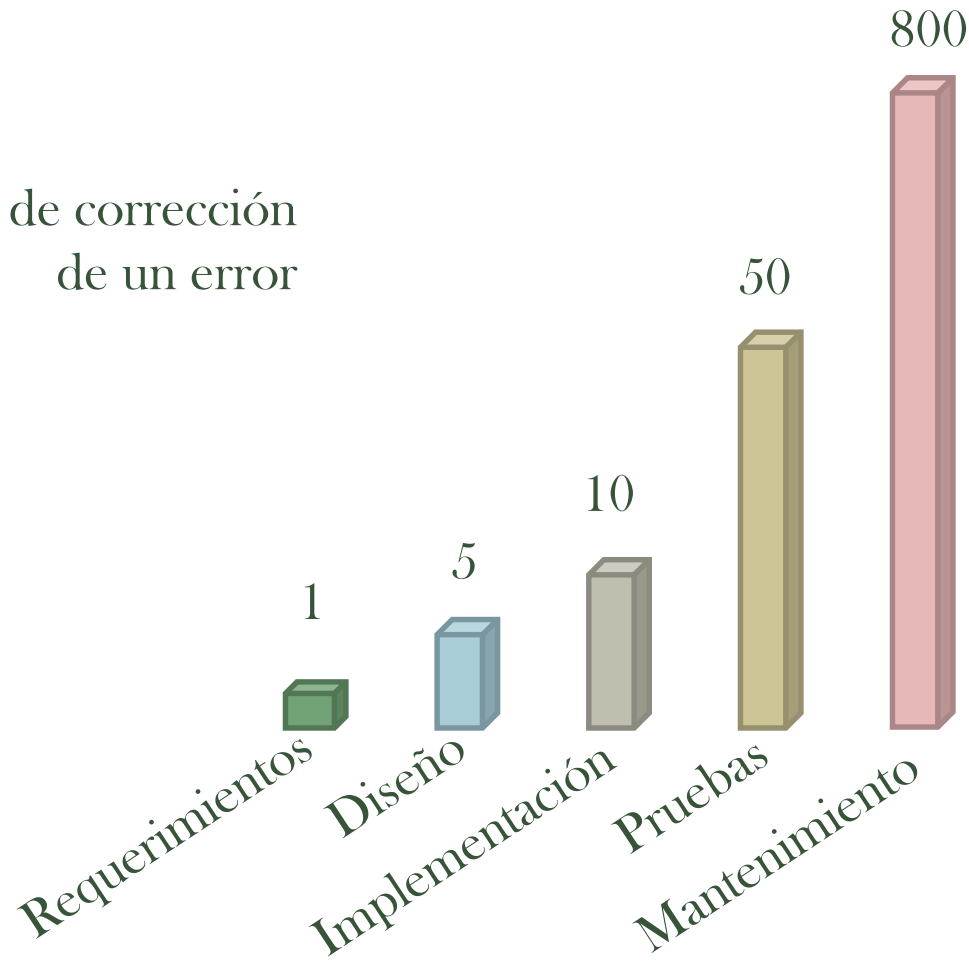
- Es el proceso realizado para establecer las necesidades de los *stakeholders* que tienen que ser resueltas por el software
- Es una etapa crucial en el proceso de desarrollo, un error en esta etapa llevara inevitablemente a un error en el diseño e implementación
- Un software puede estar desarrollado de forma perfecta, pero si no hace lo que debe hacer “no sirve”



INGENIERÍA DE REQUERIMIENTOS



Costo de corrección
de un error



INGENIERÍA DE REQUERIMIENTOS



Elicitación

Administración

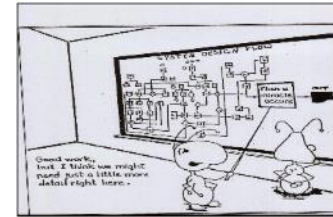
Análisis

Validación

Especificación



DISEÑO DE SOFTWARE



Actividades del diseño

Diseño
Arquitectural

Diseño de
Interfaces

Estructuras
de Datos

Especificación
Abstracta

Diseño de
Componentes

Diseño de
Algoritmos

Estructura
del sistema

Especificación
de Interfaces

Especificación
de Datos

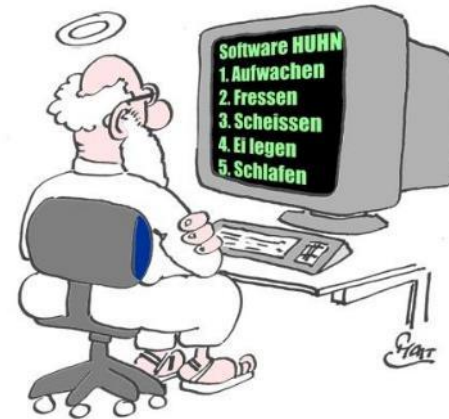
Especificación
del software

Especificación
Componentes

Especificación
de Algoritmos

Productos del diseño

IMPLEMENTACIÓN



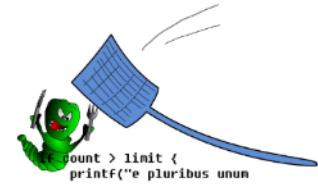
IMPLEMENTACIÓN



- Reducción de la complejidad
 - Software fácil de entender y usar
- Anticipación de la diversidad
 - Anticipar las posibles modificaciones
- Estructuración para la validación
 - Implementar para poder probar
- Uso de estándares
 - Estándares internos (nombres, identificación, etc.)
 - Estándares externos (impuestos por reglamentación)



VALIDACIÓN Y VERIFICACIÓN

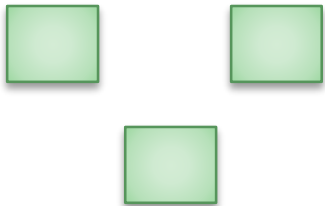


○ Validación

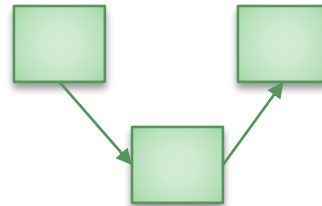
- ¿Estamos construyendo el sistema correcto?

○ Verificación

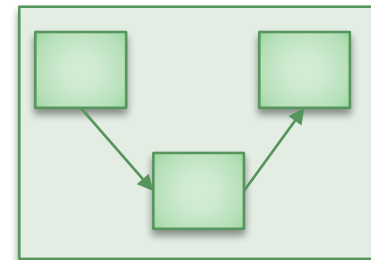
- ¿Estamos construyendo el sistema correctamente?



Unidad



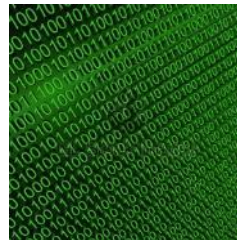
Integración



Sistema



MANTENIMIENTO



MANTENIMIENTO



- Correctivo:
 - Derivado de un reporte de errores encontrados
- Perfeccionamiento
 - Derivado de nuevas características demandadas
- Adaptativo
 - Derivado de cambios en el entorno
- Pruebas de regresión
 - Conjunto de pruebas a aplicar una vez realizada una actividad de mantenimiento

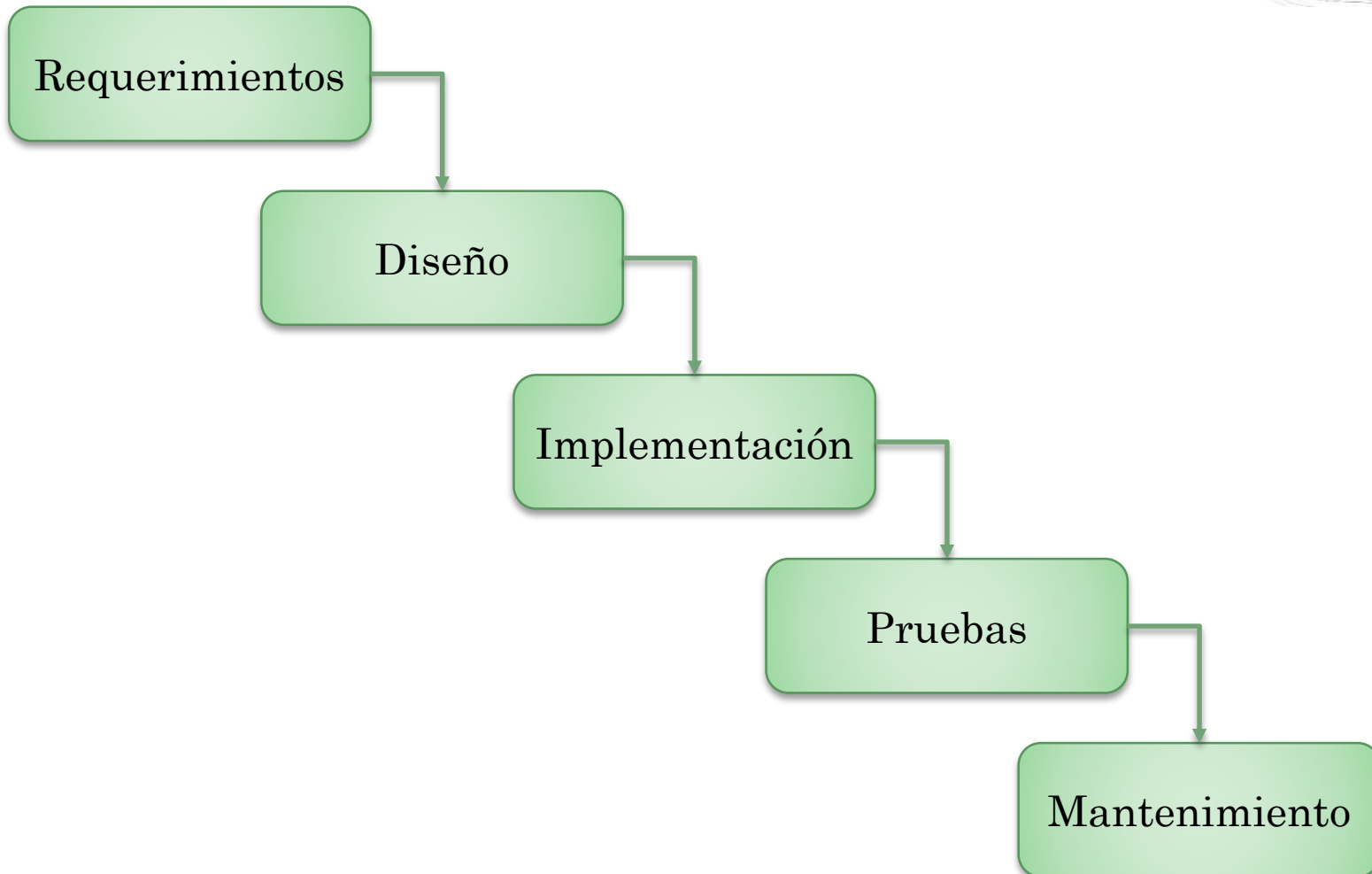


MODELO DE PROCESO SOFTWARE (O DE CICLO DE VIDA SOFTWARE)

- Como poner las actividades juntas para construir software efectivamente
 - Determinar el orden de las actividades
 - Determinar el criterio de transición entre actividades
 - Determinar los productos de las actividades
 - Determinar los roles del proyecto
- Modelos a revisar
 - Modelo Cascada
 - Modelo en Espiral
 - Modelo de Prototipos Evolutivos
 - Modelo Unificado de Desarrollo
 - Métodos Agiles*



MODELO CASCADA (WATERFALL)



MODELO CASCADA



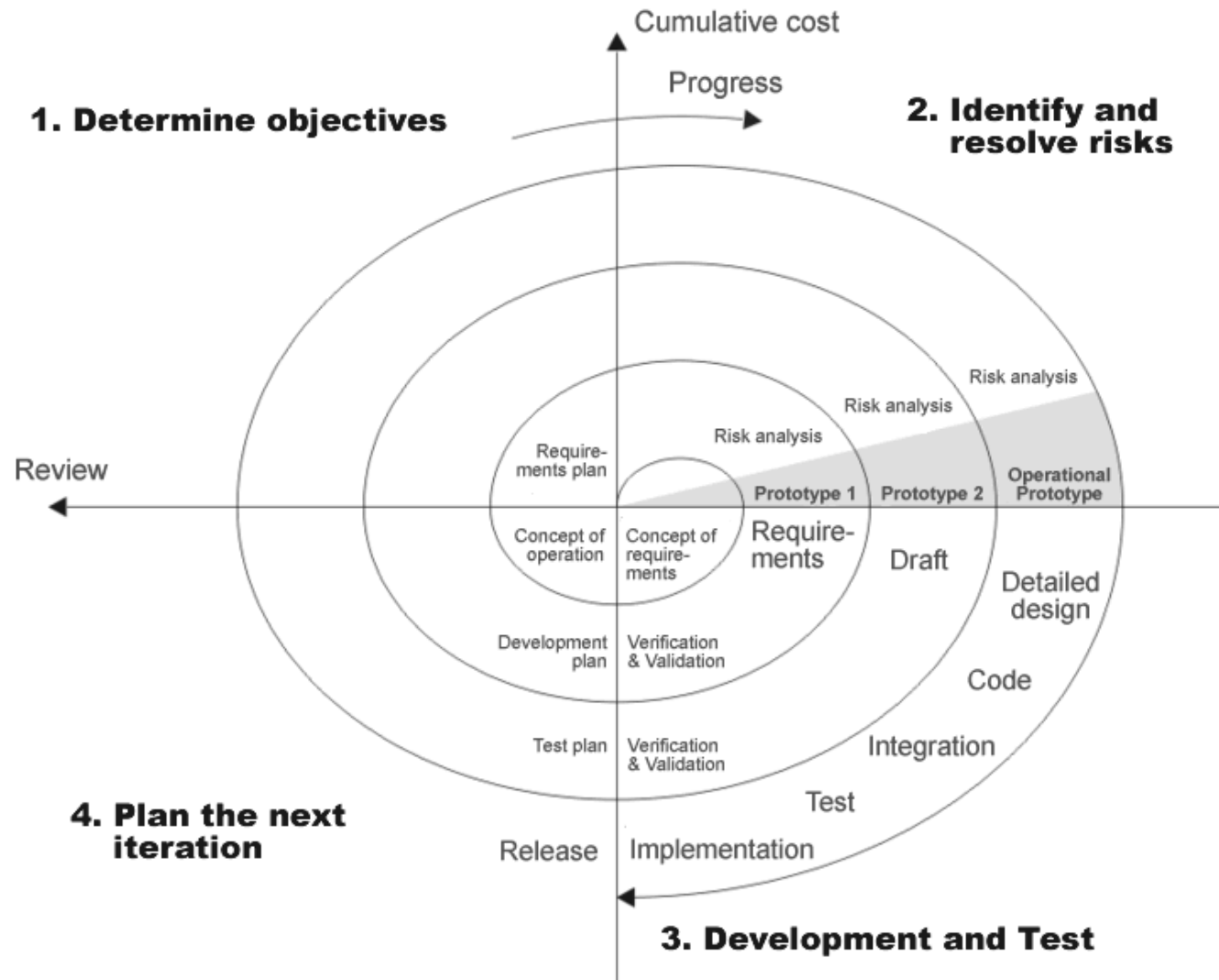
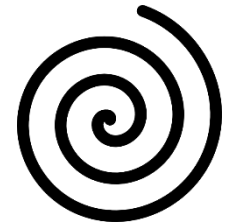
- Se adapta bien cuando
 - Definición del producto estable
 - El dominio es bien conocido
 - La tecnología se domina bien

- Ventajas
 - Permite encontrar los errores temprano
 - Fácil de seguir el avance del proyecto

- Desventajas
 - Modelo poco flexible



MODELO ESPIRAL



MODELO ESPIRAL



○ Ventajas

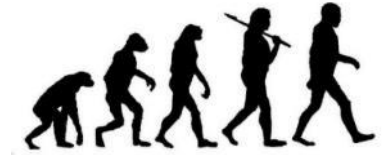
- Reducción de riesgos
- Funcionalidades pueden ser agregadas
- Prototipos ejecutables producidos de forma temprana

○ Desventajas

- El análisis de los riesgos requiere experticia
- Completamente dependiente del análisis de riesgos
- Modelo más complejo y costoso de implementar



MODELO PROTOTIPOS EVOLUTIVOS



Concepto Inicial

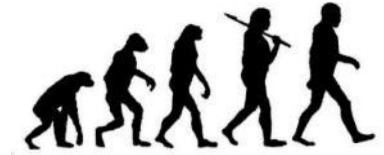
Diseño e
Implementación
del prototipo

Refinamiento del
prototipo hasta
aceptación

Completar y
entregar el
prototipo



MODELO PROTOTIPOS EVOLUTIVOS



○ Características

- Ideal cuando no se conocen o entienden bien los requerimientos
- Se empieza a desarrollar la parte mejor conocida del sistema
- El cliente ofrece retroalimentación

○ Ventajas

- Retroalimentación inmediata del cliente

○ Desventajas

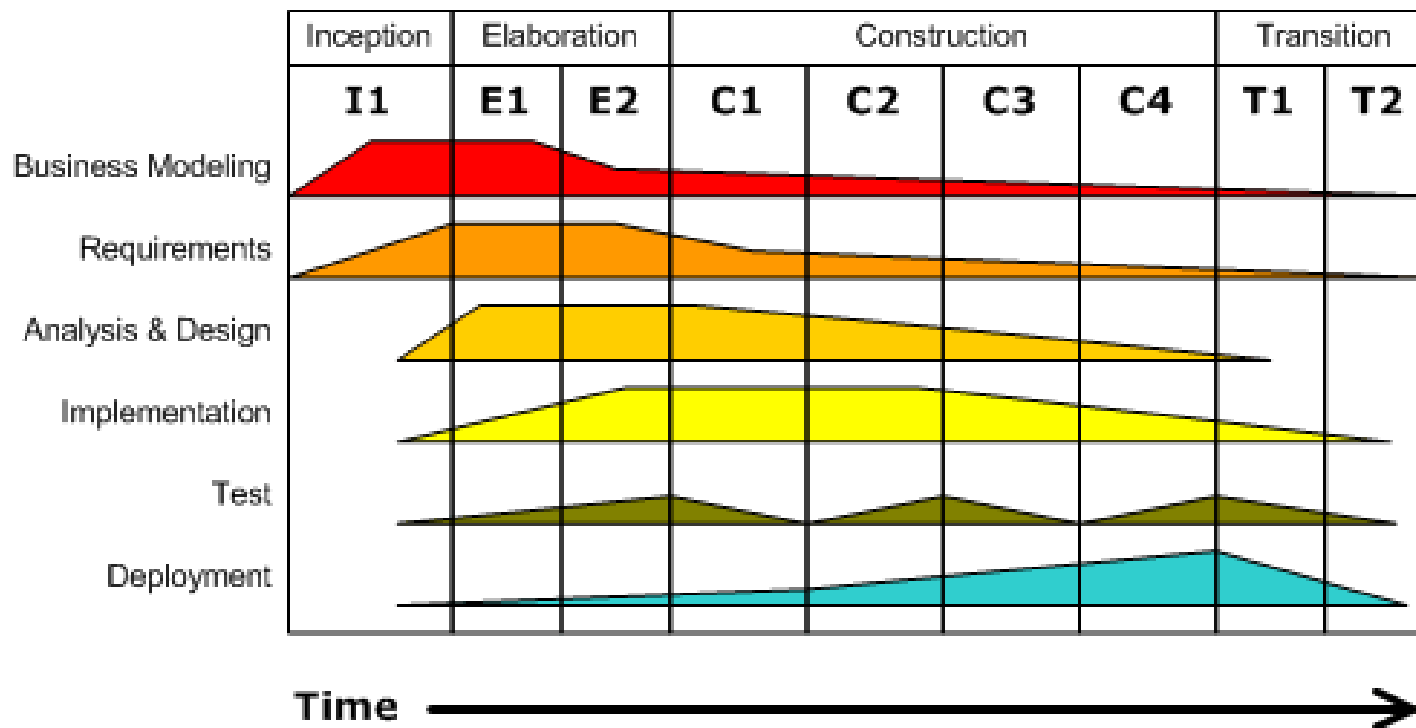
- Difícil de planear tiempo y recursos



PROCESO UNIFICADO DE DESARROLLO (RUP)

Iterative Development

Business value is delivered incrementally in time-boxed cross-discipline iterations.



QUIZ

- ¿Cual modelo escogería para implementar un software de control de un avión?
 - A. Cascada
 - B. Prototipos Evolutivos
 - C. Espiral

- ¿Cuál modelo escogería para implementar un software en cual correcciones se pueden presentar a mitad de camino?
 - A. Cascada
 - B. Prototipos Evolutivos
 - C. Espiral



DOCUMENTACIÓN

- Documentar permite comunicar entre los diferentes *stakeholders* involucrados
- Asegurar la correcta implementación del sistema
- Facilitar el mantenimiento
- Templates de documentos disponibles en IEEE



ERRORES DE PERSONAL

- Síndrome de superhéroe
 - Supuesto que una persona puede hacer de todo
- Ambiente de trabajo errado
 - Disminuye la productividad
 - Ambiente debe ser luminoso, silencioso, buen clima...
- Administración pobre
 - Falta de liderazgo
 - Liderazgo utilizado con mecanismos errados
 - Agregar personas a un proyecto retrasado



ERRORES DE PROCESO

- Problemas con cronograma
 - Incapacidad de establecer un cronograma
- Problemas con planeación
 - Planes insuficientes
 - Planes abandonados
- Fallas no vistas
 - Afecta la calidad del producto



ERRORES DE PRODUCTO

- Productos perfectos
 - Agregar características innecesarias
- Crecimiento de funcionalidad
 - Agregar más funcionalidades que las inicialmente planeados
- Proyectos I+D
 - Nuevos algoritmos y técnicas



ERRORES DE TECNOLOGÍA

- Síndrome bala plata
 - Demasiada confianza en una tecnología
- Cambiar de herramientas
 - Nuevas herramientas a mitad de camino
- No utilizar un sistema de control de versiones
 - Inhabilidad de encontrar una versión anterior



QUIZ

¿Por qué agregar personas a un proyecto retrasado no conlleva a avanzar más rápidamente?

