

# **JUSTIFICACIÓN DEL DESARROLLO DE UN SE**

- El beneficio económico que representa la solución del problema es alto
- La experiencia humana puede desaparecer
- La experiencia humana no se encuentra comúnmente disponible
- La experiencia es requerida ampliamente para la solución del problema
- La experiencia de una persona experta es requerida en ambientes hostiles

# **CARACTERÍSTICAS DE UN SE APROPIADO**

## Naturaleza de la tarea

- La tarea debe requerir manipulación simbólica
- La tarea requiere soluciones heurísticas

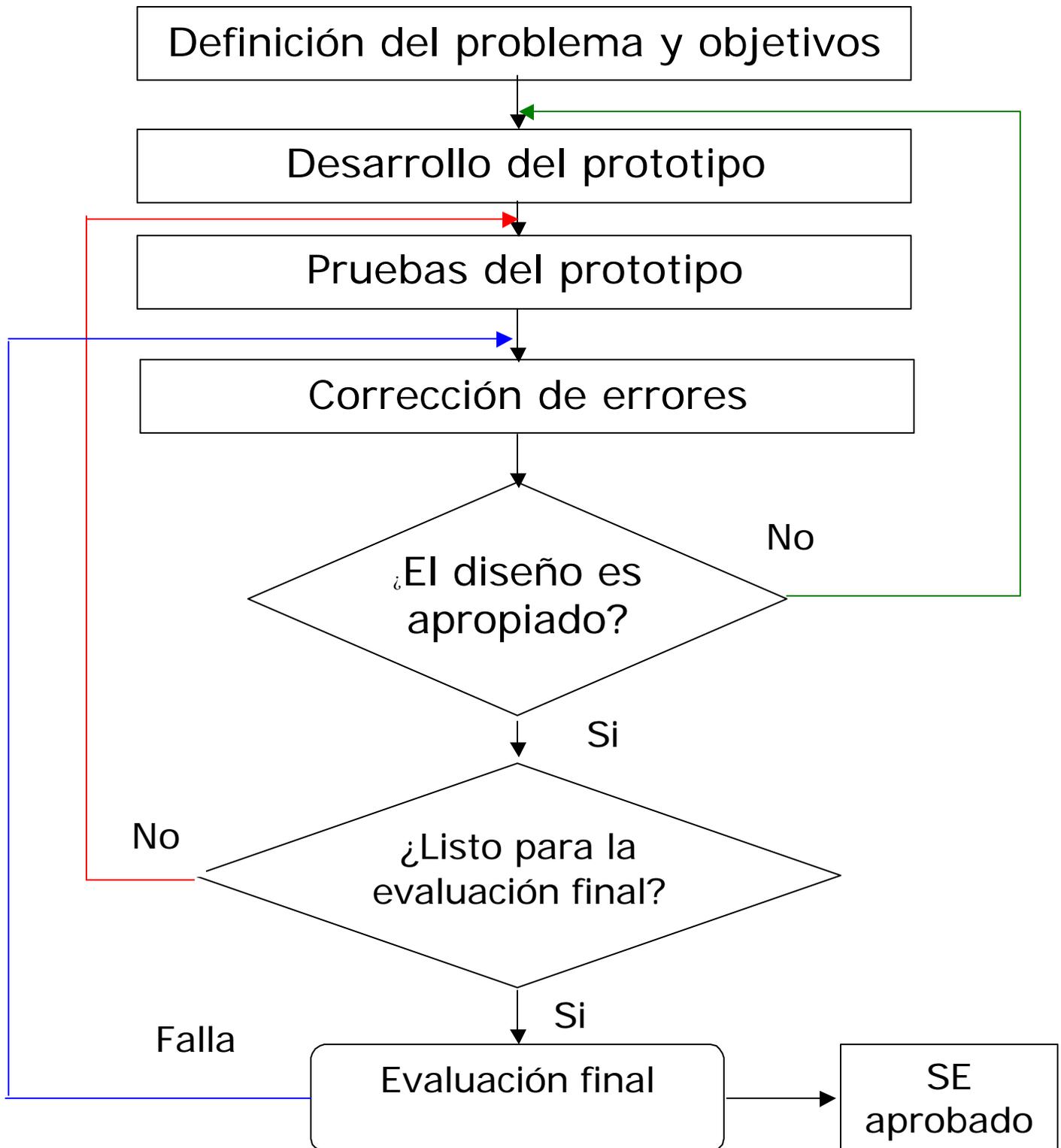
## Complejidad

- La tarea no es demasiado fácil

## Alcance

- La tarea debe tener un valor práctico
- La cantidad de información debe ser de tamaño manejable

## CICLO DE DESARROLLO DE UN SE



## **REQUERIMIENTOS PARA EL DESARROLLO DE UN SE**

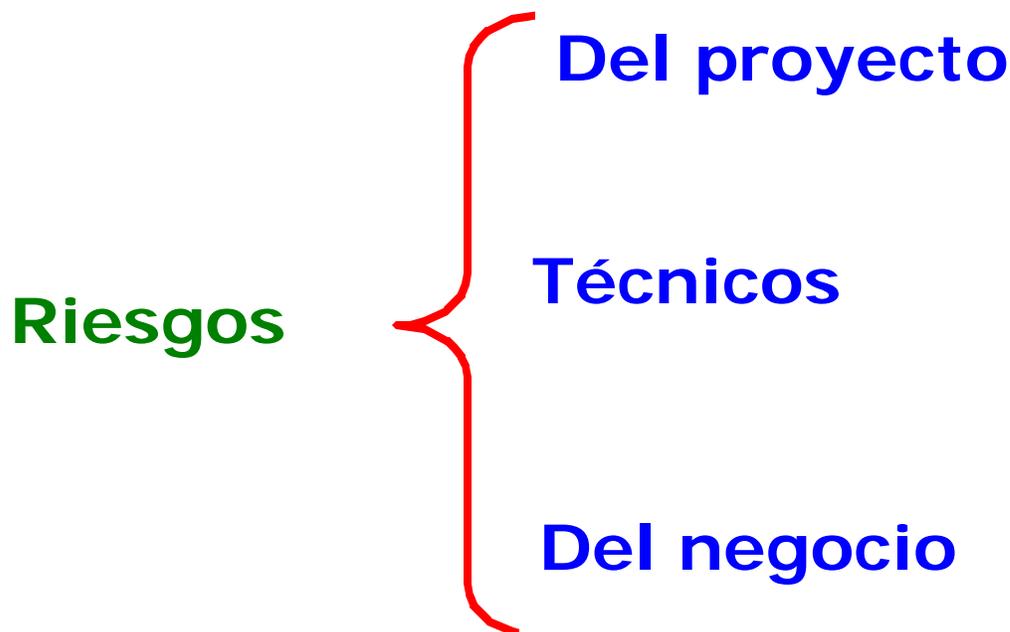
- La tarea a realizar no requiere la utilización de sentido común
- La tarea requiere únicamente habilidad cognoscitiva
- Existen expertos en el campo del problema
- Los expertos están de acuerdo en sus criterios
- Los expertos pueden describir sus métodos de análisis
- La tarea a realizar no es demasiado compleja
- La tarea no es pobremente entendida
- Intereses personales

# PLANEACIÓN DE PROYECTO DE UN SE

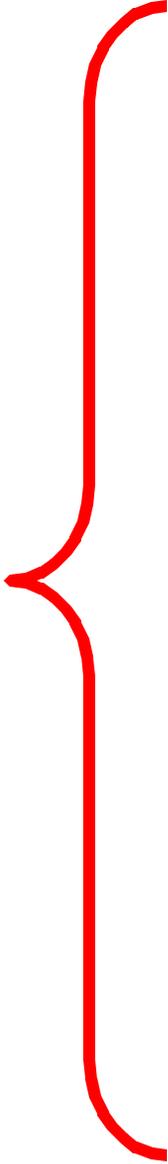
La Ingeniería de Sistemas abarca un conjunto de tres elementos:

- Métodos
- Herramientas
- Procedimientos

La planeación debe considerar un análisis de riesgo



# Riesgos del Proyecto

- 
- Presupuesto
  - Agenda
  - Personal
  - Recursos
  - Del cliente
  - De requisitos
  - Impacto de  
proyecto de  
software

## Riesgos Técnicos

- Problemas de diseño
- Implementación
- Interfaz
- Verificación
- Mantenimiento

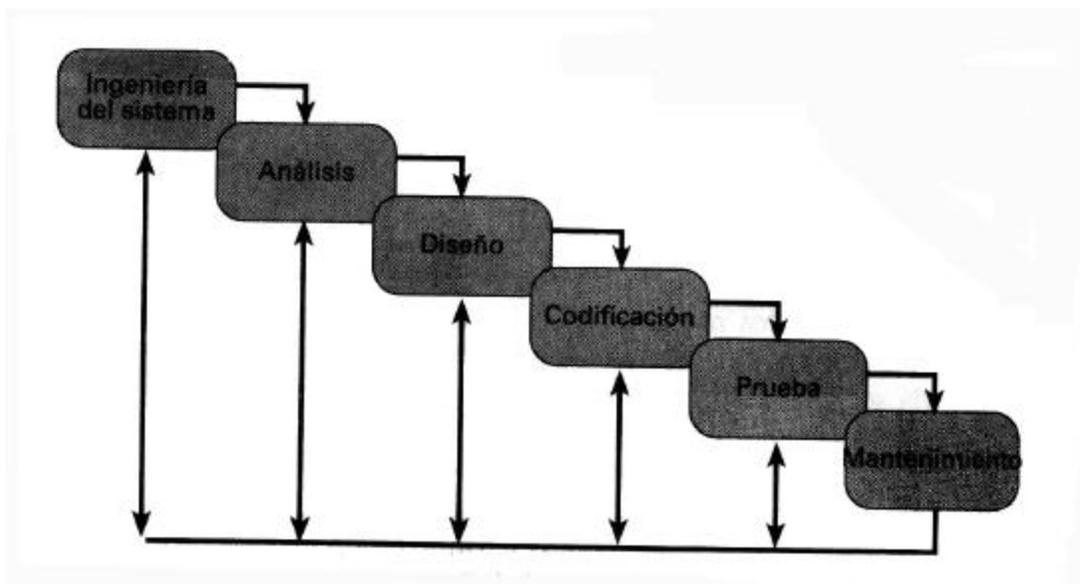
## Riesgos del Negocio

- **Producto sin demanda**
- **No se ajusta a la estrategia de la empresa**
- **No se sabe vender**
- **Pérdida del soporte de los gestores**
- **Cambio de enfoque o cambio de personal**
- **Pérdidas de presupuesto**

## EL CICLO DE VIDA CLÁSICO

Este paradigma exige un enfoque sistemático y secuencial del desarrollo del software, abarca las siguientes actividades:

- Ingeniería y análisis del sistema
- Análisis de los requisitos del software
- Diseño
- Codificación
- Prueba
- Mantenimiento



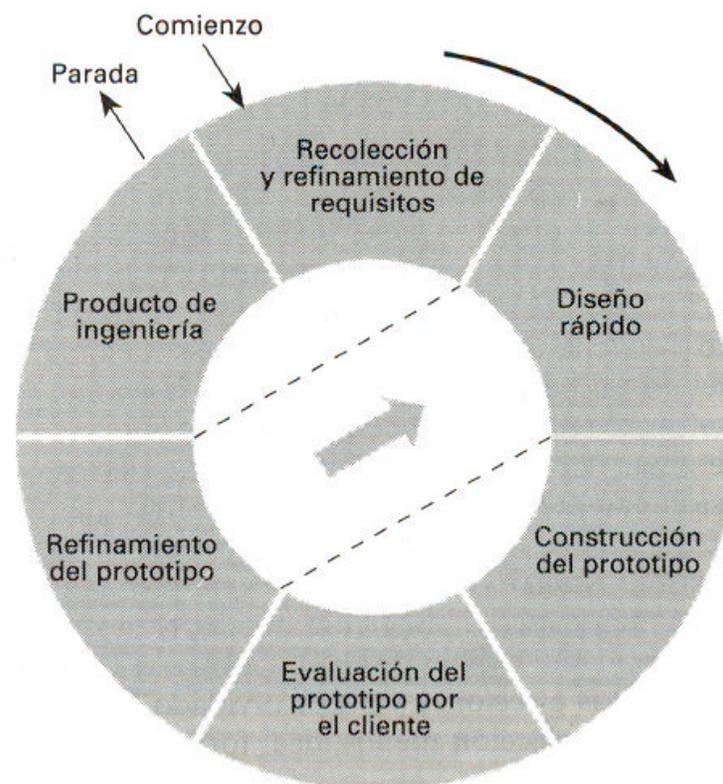
## **CONSTRUCCIÓN DE PROTOTIPOS**

Es un modelo que facilita al programador la creación de un modelo del software a construir.

Secuencia de sucesos del paradigma de construcción de prototipos:

- **Recolección de los requisitos:** El técnico y el cliente definen los objetivos globales para el software e identifican los requisitos conocidos.
- **Diseño rápido:** Se enfoca sobre la representación de los aspectos del software visibles al usuario.
- **Construcción del prototipo**

- Evaluación del prototipo por el cliente
- Refinamiento del prototipo: Se produce un proceso interactivo en el que el prototipo es afinado para que satisfaga las necesidades del cliente.
- Producto de ingeniería

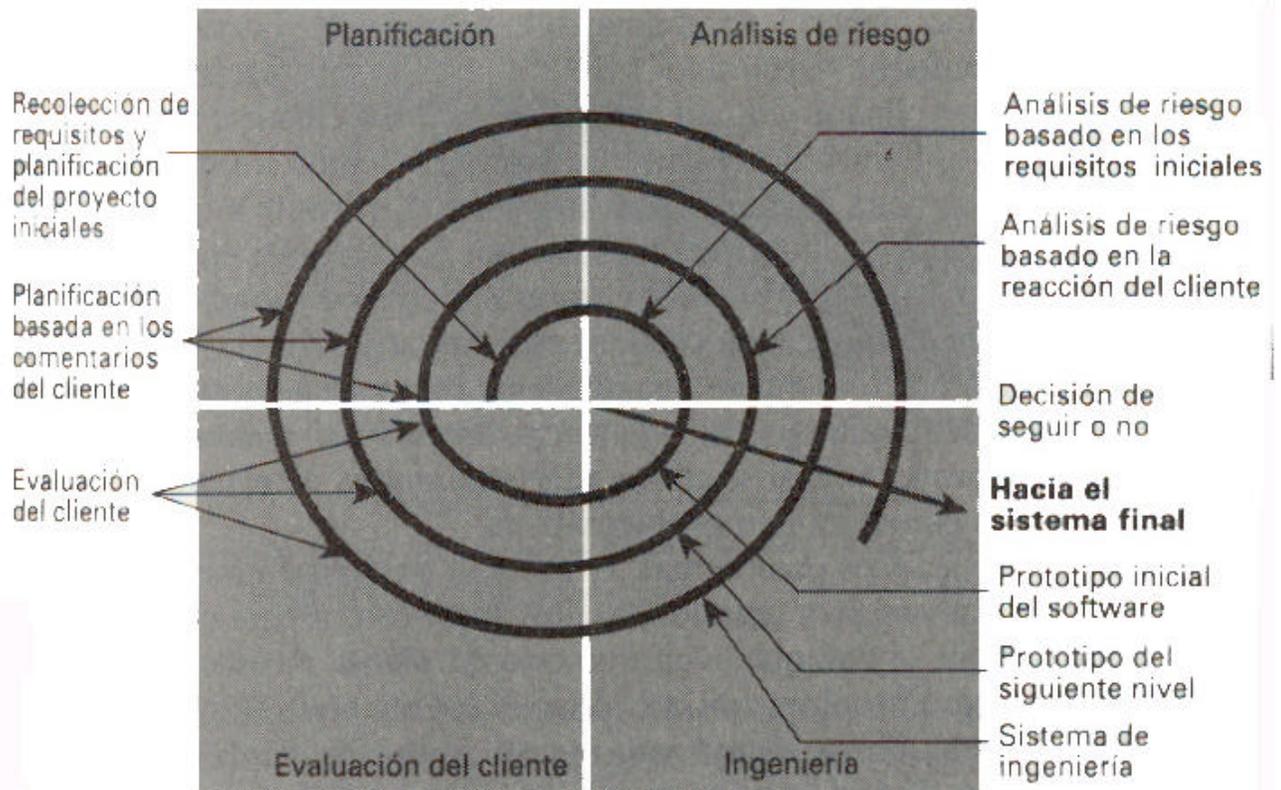


## EL MODELO EN ESPIRAL

Este modelo cubre las mejores características tanto del ciclo de vida clásico, como de la creación de prototipos, añadiendo un nuevo elemento: el análisis de riesgo.

Las cuatro actividades principales del modelo en espiral son:

- Planificación: determinación de objetivos, alternativas y restricciones
- Análisis de riesgo: análisis de alternativas de identificación y resolución de riesgos
- Ingeniería: desarrollo del producto de "siguiente nivel"
- Evaluación del cliente: valoración de los resultados de la ingeniería



Con cada iteración alrededor de la espiral (comenzando en el centro y siguiendo hacia el exterior), se construyen sucesivas versiones del software, cada vez mas completas.

# GESTION DEL PROYECTO: ESTIMACIÓN

La estimación de recursos, costes y agendas para el esfuerzo de desarrollo de software requiere:

- Experiencia
- Acceso a una buena información histórica
- Coraje para confiar en medidas cuantitativas cuando todo lo que existe son datos cualitativos

