

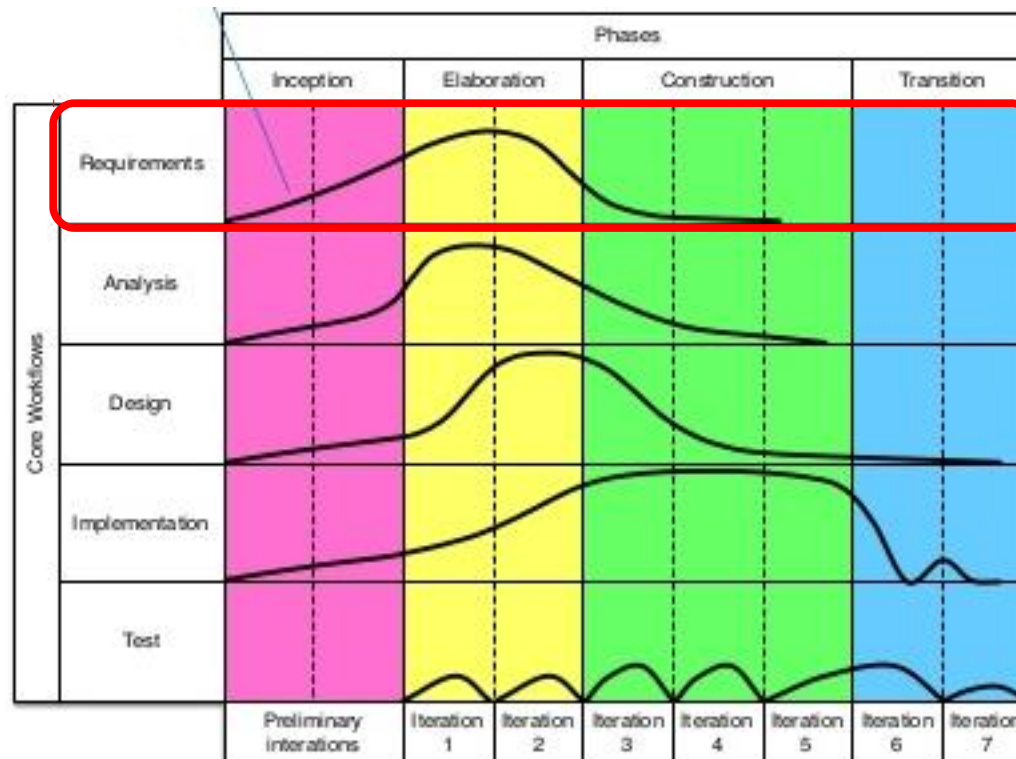
# WORK FLOW: REQUISITOS INTRODUCCIÓN

DISEÑO Y CONSTRUCCIÓN DE SOFTWARE

# Propósito

## ■ Captura de requisitos

- ◆ Proceso de averiguar lo que se debe construir
- ◆ Mayor importancia al principio del proyecto



# Propósito

## ■ Captura de requisitos

- ◆ Crea una especificación de alto nivel sobre lo que el sistema debería de hacer
- ◆ Preguntar a todos los interesados
  - Usuarios, mantenimiento, personal apoyo, ventas, ...
  - Priorizar los requerimientos
  - Llegar a un acuerdo
- ◆ Condiciones y capacidades que el sistema debe cumplir
  - Utilizar el *lenguaje del cliente*

# Visión general

## ■ Pasos

- ◆ Enumerar requisitos candidatos
- ◆ Comprender el contexto del sistema
- ◆ Capturar requisitos funcionales
- ◆ Capturar requisitos no funcionales



# Visión general

- Enumerar requisitos candidatos
  - ◆ Creación de una lista de características que podría tener el sistema (buenas ideas)
    - Se utiliza para planificar el trabajo
    - Puede que se conviertan en requisitos o no
    - Incluirse en versiones futuras del sistema
  - ◆ Información sobre cada característica
    - Nombre corto,
    - Definición,
    - Conjunto de valores de planificación
      - Estado, coste estimado de implementación, prioridad, nivel de riesgo

# Visión general

- Comprender el contexto del sistema
  - ◆ Modelado del dominio
    - Describir los conceptos importantes del contexto
    - Relacionarlos unos con otros
    - Creación de un glosario de términos
  - ◆ Modelado del negocio
    - Describir los procesos existentes u observados
    - Cuáles se soportarán
    - Competencias requeridas por cada uno

# Visión general

- Captura de requisitos funcionales
  - ◆ Afirmaciones sobre lo que el sistema debería hacer
  - ◆ Afirmaciones sobre la funcionalidad del sistema
- Captura de requisitos no funcionales
  - ◆ Propiedades del sistema
    - Restricciones del entorno, rendimiento, ....

# Descripción

## ■ Artefacto

- ◆ Elemento de información tangible que es creada, modificada y usada por los trabajadores

## ■ Trabajador

- ◆ Puesto al que se le puede asignar una persona real
- ◆ Tiene asignadas unas responsabilidades y el comportamiento esperado

## ■ Flujo de trabajo

- ◆ Diagrama que incluye a los trabajadores, las actividades que realizan y los artefactos que producen



# Artefactos

- Modelo de casos de uso
  - ◆ Diferentes casos de uso
  - ◆ Actores
  - ◆ Relaciones
- Modelo de requisitos
  - ◆ Requisitos funcionales
  - ◆ Requisitos no funcionales
- Descripción de la arquitectura
  - ◆ Vista de la arquitectura del modelo de casos de uso
  - ◆ Casos de uso críticos

# Artefactos

## ■ Glosario

- ◆ Términos comunes importantes que se utilizan al describir el sistema
- ◆ Consenso en las definiciones de conceptos y nociones
- ◆ Reduce el riesgo de confusiones

## ■ Prototipo de interfaz de usuario

- ◆ Ayuda a comprender y especificar la interacciones entre los actores y el sistema

# Trabajadores

## ■ Analista de sistemas

### ◆ Responsable de

- Modelo de casos de uso
  - Completo y consistente
- Actores
- Glosario

### ◆ No es responsable de cada caso de uso en particular

## ■ Especificador de casos de uso

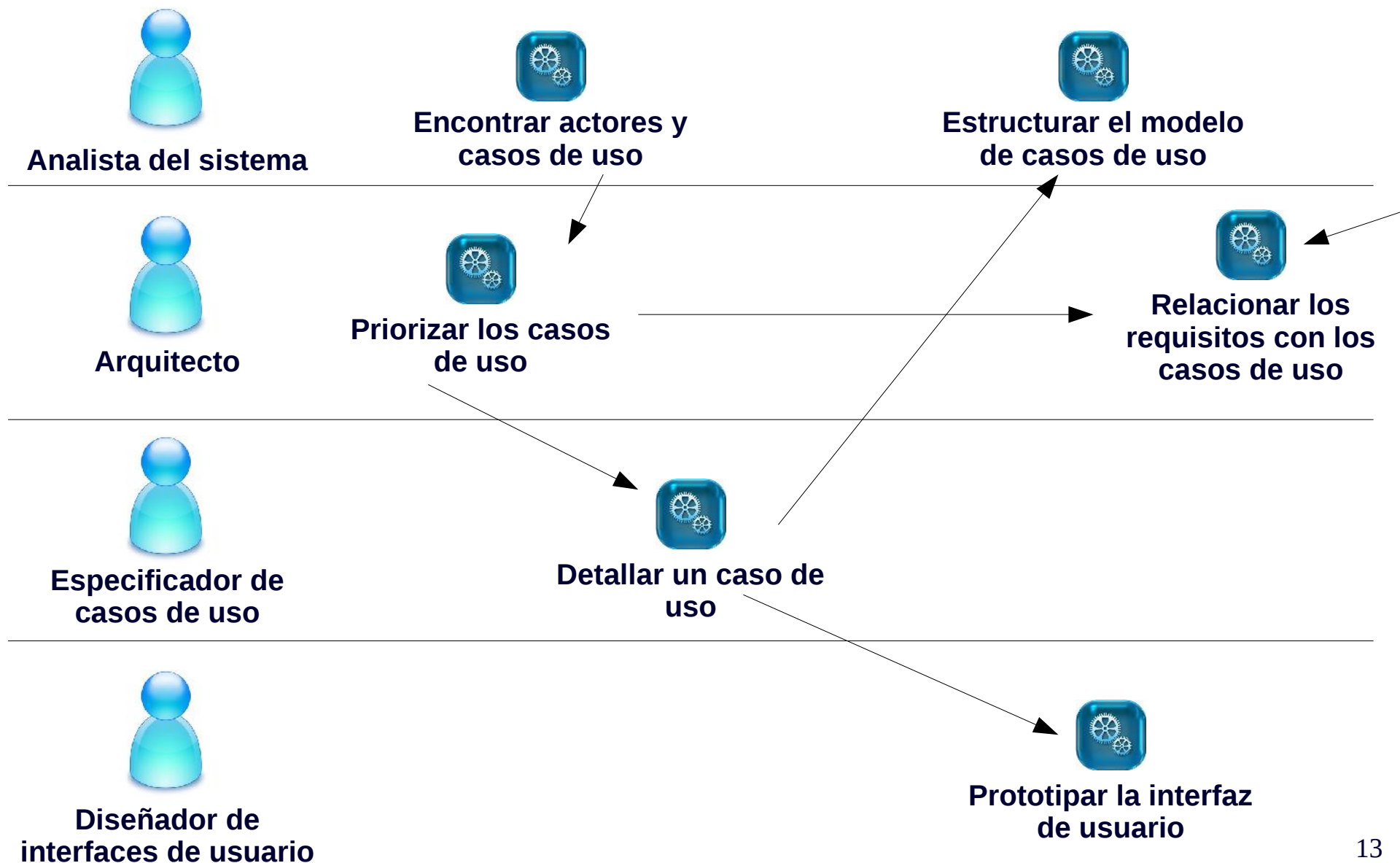
### ◆ Responsable de:

- Descripción detallada de uno o más casos de uso

# Trabajadores

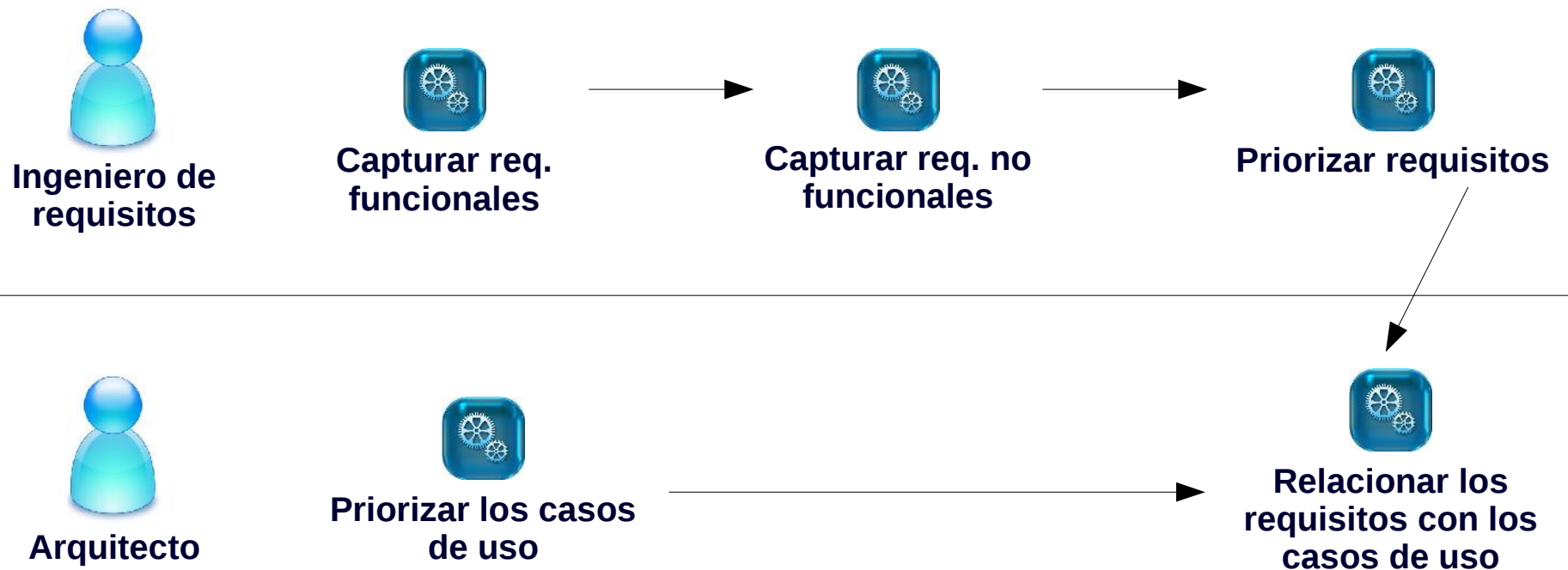
- Ingeniero de requisitos
  - ◆ Responsable de
    - Capturar los requisitos funcionales y no funcionales
    - Priorizar los requisitos
- Arquitecto
  - ◆ Responsable de
    - Priorizar los casos de uso
    - Relacionar los requisitos con los casos de uso
- Diseñador de interfaz de usuario
  - ◆ Responsable de
    - Prototipar la interfaz de usuario

# Flujo de trabajo





# Flujo de trabajo

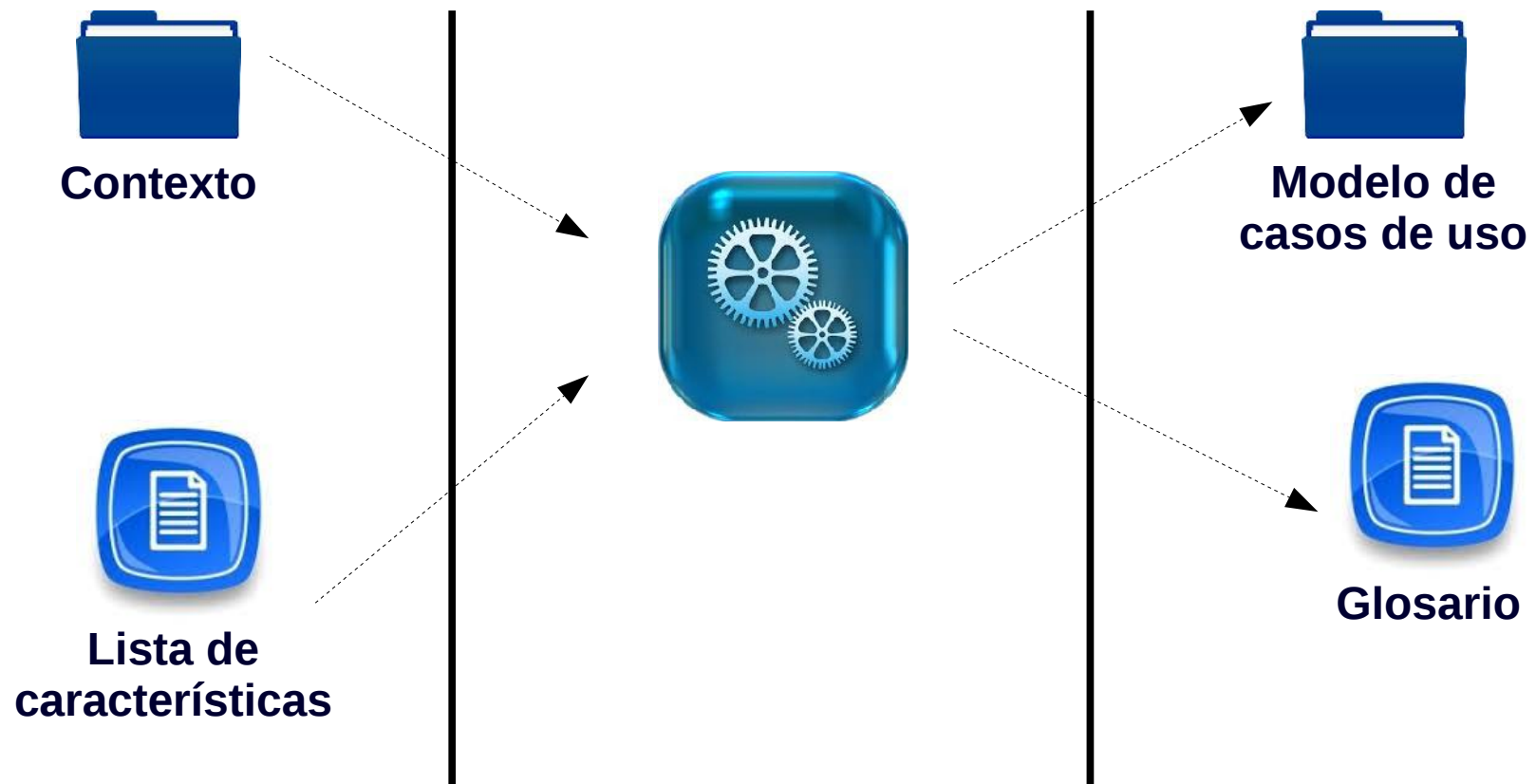


# Actividades

- Encontrar actores y casos de uso
  - ◆ Delimitar el sistema de su entorno
  - ◆ Esbozar quién y qué interactuarán con el sistema
    - Actores
  - ◆ Esbozar qué funcionalidad se espera del sistema
    - Casos de uso
  - ◆ Capturar y definir un glosario de términos comunes
    - Descripciones detalladas de las funcionalidades del sistema

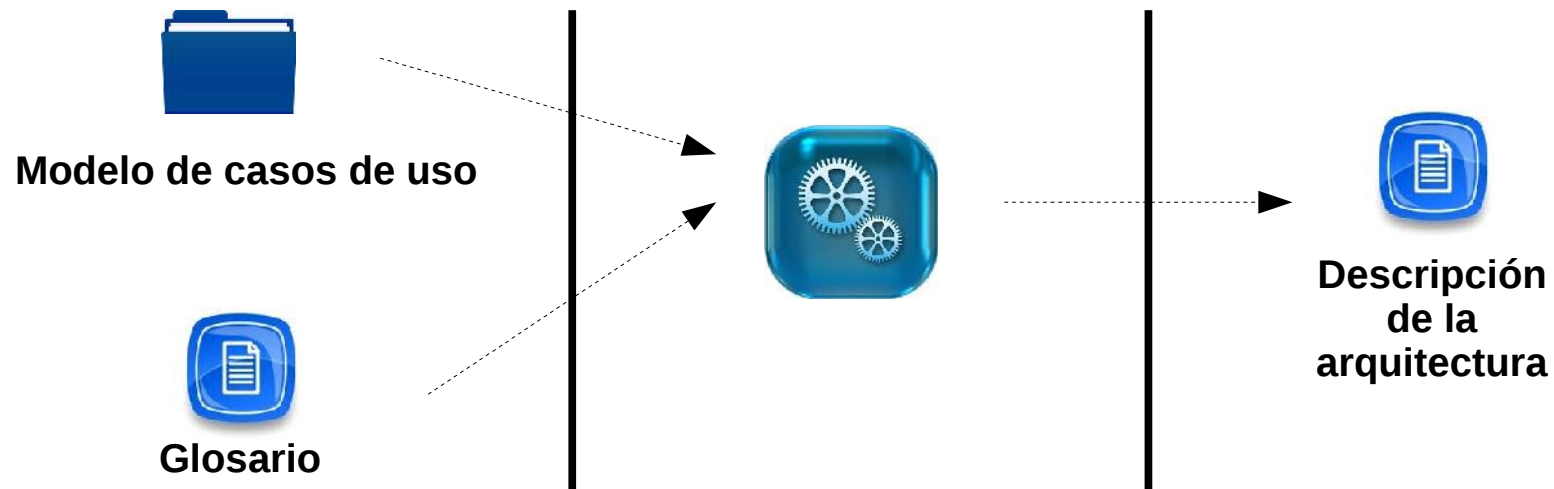
# Actividades

## ■ Encontrar actores y casos de uso



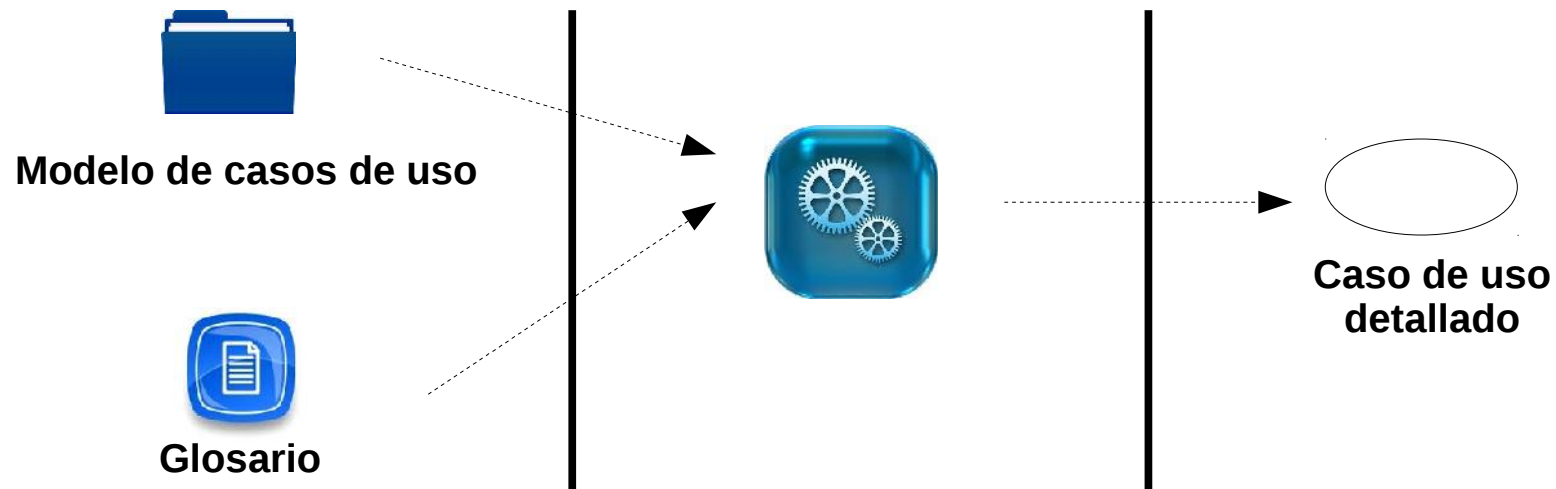
# Actividades

- Priorizar casos de uso
  - ◆ Cuáles son necesarios desarrollar en la primeras iteraciones
    - Análisis, diseño e implementación



# Actividades

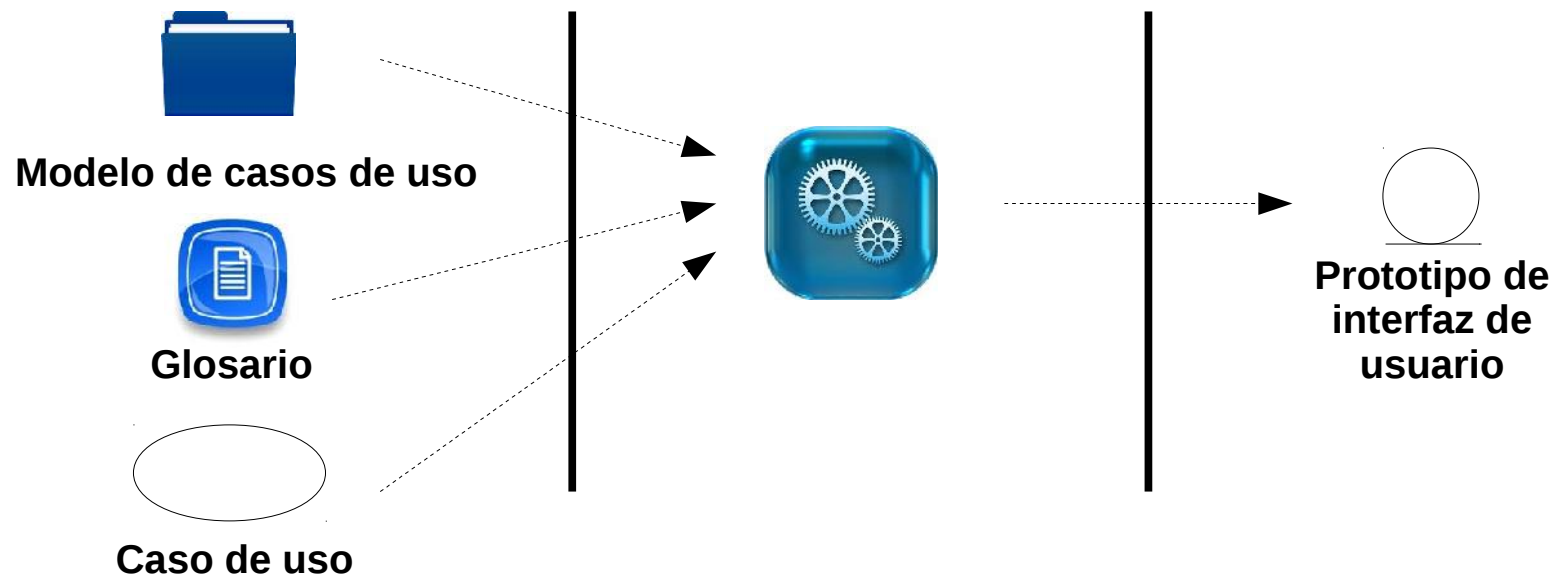
- Detallar casos de uso
  - ◆ Describir flujo de sucesos en detalle
    - Cómo comienza, termina e interactúan los usuarios
    - En formato texto y diagrama





# Actividades

- Prototipar la interfaz de usuario
  - ◆ Construir un prototipo de la interfaz del usuario
    - Llevar a cabo los casos de uso de manera eficiente
    - Esquema de interfaces de usuario y prototipos
      - Apariencia de las interfaces de cara a los usuarios

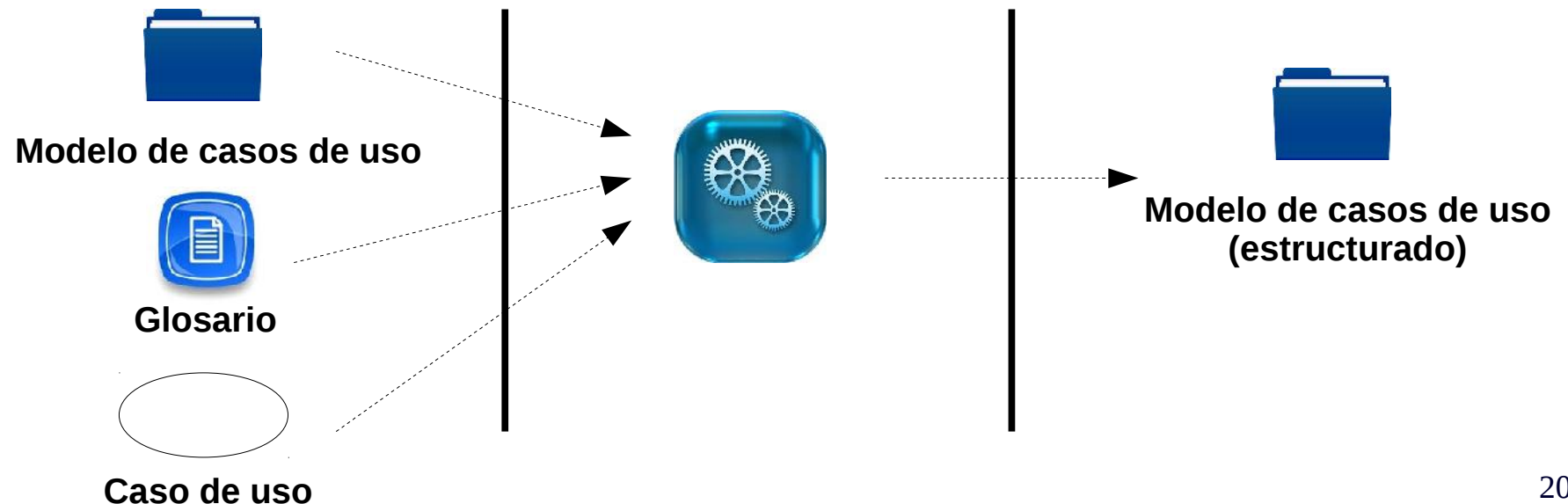


# Actividades

## ■ Estructurar el modelo de casos de uso

### ◆ Extraer

- Descripciones de funcionalidad generales y compartidas
  - Casos de uso más específicos
- Descripciones adicionales u opcionales
  - Extender las descripciones de los casos de uso



# Especificación de requisitos software (SRS)

- Fase más temprana del proceso de construcción del software
  - ◆ Se utiliza como entrada para el análisis y el diseño
- Se utilizan dos modelos diferentes
  - ◆ Modelo de requisitos
    - Requisitos funcionales
    - Requisitos no funcionales
  - ◆ Modelo de casos de uso
    - Diferentes casos de uso
    - Actores
    - Relaciones

