

# Técnicas de Recolección de Requerimientos

Análisis y Diseño I

Undécimo A

03 de agosto del 2025

La fase de análisis en el desarrollo de software es crucial, y en su corazón se encuentra la recolección de requerimientos. Este proceso es fundamental para entender qué necesita el usuario y cómo el sistema debe satisfacer esas necesidades. Una recolección de requerimientos deficiente es una de las principales causas de fracaso en los proyectos de software.

## ¿Qué son los Requerimientos?

Los **requerimientos** son las condiciones o capacidades que debe poseer un sistema para satisfacer un contrato, una norma, una especificación u otros documentos formalmente impuestos. En términos más simples, son las necesidades y expectativas de los usuarios y de las partes interesadas (stakeholders) que el software debe cumplir.

Se clasifican principalmente en dos tipos:

1. **Requerimientos Funcionales:** Describen lo que el sistema *debe hacer*. Son las funciones o servicios que el sistema proporcionará.
  - Ejemplos: "El sistema debe permitir a los usuarios registrarse con un correo electrónico y contraseña." "El sistema debe generar reportes de ventas mensuales."
2. **Requerimientos No Funcionales:** Describen cómo el sistema *debe ser*. Son atributos de calidad, restricciones o criterios que el sistema debe cumplir.
  - Ejemplos: "El sistema debe ser capaz de soportar 100 usuarios concurrentes." "El tiempo de respuesta de la página de inicio no debe exceder los 3 segundos." "El sistema debe ser seguro contra ataques de inyección SQL."

## Proceso de Recolección de Requerimientos

La recolección de requerimientos es un proceso iterativo que generalmente sigue estos pasos:

1. **Identificación de Stakeholders:** Determinar quiénes son las personas o grupos afectados por el sistema o que tienen interés en él.
2. **Definición del Alcance:** Establecer los límites del proyecto y qué funcionalidades se incluirán o excluirán.
3. **Selección de Técnicas:** Elegir las técnicas más adecuadas para obtener la información de los stakeholders.
4. **Ejecución de la Recolección:** Aplicar las técnicas seleccionadas (entrevistas, encuestas, etc.).
5. **Documentación de Requerimientos:** Registrar los requerimientos de forma clara, concisa y sin ambigüedades.
6. **Validación y Verificación:** Confirmar con los stakeholders que los requerimientos documentados son correctos y completos.

# Técnicas Comunes de Recolección de Requerimientos

Existen diversas técnicas que los analistas de sistemas utilizan para extraer información de los usuarios y otras fuentes:

## 1. Entrevistas

Conversaciones directas con los stakeholders para entender sus necesidades, procesos y expectativas. Pueden ser estructuradas (con preguntas predefinidas) o no estructuradas (más abiertas).

- **Ventajas:** Permite obtener información detallada, clarificar dudas al instante, observar el lenguaje corporal.
- **Desventajas:** Consume tiempo, puede estar sujeta a sesgos, requiere habilidades de comunicación del entrevistador.

## 2. Cuestionarios / Encuestas

Conjunto de preguntas escritas distribuidas a un gran número de stakeholders. Útiles para recopilar información cuantitativa o cuando los stakeholders están dispersos geográficamente.

- **Ventajas:** Eficiente para grandes grupos, fácil de analizar (si son preguntas cerradas), permite anonimato.
- **Desventajas:** No permite aclaraciones en tiempo real, baja tasa de respuesta, no capta matices.

## 3. Observación

El analista observa a los usuarios mientras realizan sus tareas diarias en su entorno de trabajo. Permite entender los procesos tal como realmente ocurren, no como se describen.

- **Ventajas:** Obtiene información de primera mano, identifica problemas no mencionados, valida información de otras técnicas.
- **Desventajas:** Puede ser intrusiva, los usuarios pueden cambiar su comportamiento al ser observados (efecto Hawthorne), consume tiempo.

## 4. Análisis de Documentos

Revisión de la documentación existente de la organización (manuales de procedimientos, reportes, formularios, diagramas de flujo). Proporciona una base para entender los procesos actuales y la terminología.

- **Ventajas:** Fuente de información formal y objetiva, útil para entender el "as-is" (estado actual).
- **Desventajas:** Los documentos pueden estar desactualizados o incompletos, no revela necesidades futuras.

## 5. Prototipos

Creación de una versión simplificada o parcial del sistema para que los usuarios puedan interactuar con ella y proporcionar retroalimentación temprana. Puede ser de baja fidelidad (bocetos) o alta fidelidad (interactivo).

- **Ventajas:** Clarifica los requerimientos, reduce malentendidos, obtiene feedback visual y tangible, involucra al usuario.
- **Desventajas:** Los usuarios pueden esperar el sistema final, puede ser costoso y consumir tiempo si no se gestiona bien.

## 6. Sesiones de Lluvia de Ideas (*Brainstorming*)

Reuniones grupales donde los participantes generan ideas de forma libre y espontánea sobre las funcionalidades o soluciones del sistema.

- **Ventajas:** Genera muchas ideas rápidamente, fomenta la creatividad y la colaboración.
- **Desventajas:** Puede ser dominada por pocas personas, algunas ideas pueden ser inviables, requiere un facilitador eficaz.

## 7. Casos de Uso

Descripción de las interacciones entre un usuario (actor) y el sistema para lograr un objetivo específico. Ayudan a definir los requerimientos funcionales desde la perspectiva del usuario.

- **Ventajas:** Centrado en el usuario, fácil de entender para stakeholders no técnicos, ayuda a identificar escenarios de éxito y error.
- **Desventajas:** Puede ser laborioso para sistemas muy complejos, no cubre bien los requerimientos no funcionales.

## Consejos para una Recolección Exitosa

- **Involucra a los Stakeholders Correctos:** Asegúrate de hablar con todos los grupos relevantes.
- **Escucha Activamente:** Presta atención no solo a lo que se dice, sino a lo que no se dice.
- **Haz Preguntas Abiertas y Cerradas:** Combina para obtener detalles y confirmaciones.
- **Sé Neutral:** Evita influir en las respuestas de los usuarios.
- **Documenta Claramente:** Usa un lenguaje preciso y evita la ambigüedad.
- **Valida Constantemente:** Confirma los requerimientos con los stakeholders a medida que los recolectas.
- **Prioriza:** No todos los requerimientos tienen la misma importancia.

## ¿Sabías que...?

- Uno de los errores más comunes en proyectos de software es la "deriva de alcance" (scope creep), que ocurre cuando los requerimientos cambian o aumentan sin control después de que el proyecto ha comenzado. Una buena recolección ayuda a mitigar esto.
- El "Modelo de Requerimientos" es un artefacto clave en el análisis de sistemas, y puede incluir diagramas de casos de uso, prototipos, modelos de datos y descripciones textuales.
- La técnica "JAD" (Joint Application Development - Desarrollo Conjunto de Aplicaciones) es un enfoque intensivo que reúne a usuarios y desarrolladores en talleres estructurados para definir rápidamente los requerimientos.
- Se estima que un error en la fase de requerimientos es hasta 100 veces más costoso de corregir en la fase de mantenimiento que si se hubiera detectado en la fase inicial.

# Ejercicios Básicos Prácticos

## 1. Identificación de Requerimientos:

- Para un sistema de gestión de biblioteca, identifica al menos 3 requerimientos funcionales y 3 requerimientos no funcionales.
- Ejemplos:
  - *Funcional*: "El sistema debe permitir buscar libros por título, autor y género."
  - *No Funcional*: "El sistema debe ser accesible para usuarios con discapacidades visuales."

## 2. Selección de Técnica:

- Imagina que estás desarrollando una aplicación móvil para pedir comida a domicilio. ¿Qué técnica de recolección de requerimientos considerarías más efectiva para entender las preferencias de los usuarios finales y por qué? Justifica tu respuesta.

# Análisis o Reflexión Final

La recolección de requerimientos es el cimiento sobre el cual se construye cualquier sistema de software exitoso. No es simplemente un paso inicial, sino un proceso continuo de descubrimiento y refinamiento. Un analista de sistemas debe ser un detective, un comunicador y un facilitador, utilizando una variedad de técnicas para desenterrar las verdaderas necesidades y expectativas. Ignorar esta fase o realizarla de manera superficial conduce inevitablemente a productos que no satisfacen a sus usuarios, generando frustración y costos adicionales. Invertir tiempo y esfuerzo en una recolección de requerimientos exhaustiva y precisa es la mejor garantía para desarrollar un software que realmente agregue valor y cumpla con su propósito.