

# Tecnología en el Diseño de Servicios basado en ITIL v3

Drossos Thanos

26 de octubre de 2025

## Resumen

El capítulo 5 del manual ITIL v3 (Office of Government Commerce (OGC), 2011) aborda en detalle las actividades tecnológicas clave en el diseño de servicios, centrándose especialmente en tres áreas fundamentales: la ingeniería de requerimientos, la gestión de datos e información, y la gestión de aplicaciones. Se destaca la importancia de documentar claramente requerimientos funcionales, operativos y de usabilidad mediante técnicas rigurosas como entrevistas, talleres y prototipado. Asimismo, se enfatiza la gestión estratégica y operativa de los datos, considerando su clasificación, valoración y ciclo de vida para asegurar su integridad y relevancia. Finalmente, se subraya el rol de la gestión de aplicaciones mediante frameworks, patrones de diseño y herramientas CASE, que garantizan la alineación entre las necesidades empresariales y las soluciones tecnológicas implementadas.

## Índice

<b>1. Ingeniería de Requisitos</b>	<b>2</b>
1.1. Tipos de requisitos . . . . .	2
1.2. Técnicas de investigación de requisitos . . . . .	2
1.3. Problemas comunes y soluciones en la ingeniería de requisitos . . . . .	3
1.4. Conocimiento tácito y explícito en los requisitos . . . . .	4
<b>2. Outsourcing en la Ingeniería de Requisitos</b>	<b>4</b>
<b>3. Gestión de Datos e Información</b>	<b>5</b>
3.1. Ciclo de vida de los datos . . . . .	6
3.2. Clasificación y estándares de la información . . . . .	6
3.3. Propiedad de los datos . . . . .	6
<b>Referencias</b>	<b>6</b>

# 1. Ingeniería de Requisitos

La ingeniería de requisitos es el proceso sistemático para comprender y documentar rigurosamente las necesidades del negocio, de los usuarios y de otros grupos de interés en relación con un servicio de TI. Su objetivo es asegurar que todos los requisitos se identifiquen con suficiente detalle, se analicen y validen adecuadamente, manteniendo la trazabilidad de cada requisito a lo largo de los cambios. Este proceso abarca típicamente las etapas de **elicitación** (obtención de requisitos), **análisis** (refinamiento y retroalimentación sobre lo obtenido) y **validación** (verificación de que los requisitos son completos y correctos). El resultado principal es un documento de requisitos riguroso y completo, a menudo organizado como un catálogo de requisitos individuales gestionado como parte del portafolio de servicios de TI. Existen guías y estándares reconocidos para apoyar esta ingeniería, tales como IEEE 830 (especificaciones de requisitos de software) o marcos de trabajo como SWEBOK y CMMI, que aportan buenas prácticas.

## 1.1. Tipos de requisitos

En el contexto de diseño de servicios, los requisitos se pueden clasificar en tres grandes categorías:

- **Requisitos funcionales:** Definen las funciones o tareas específicas que el servicio debe realizar, es decir, la utilidad que proporcionará al usuario o al negocio. Representan las características y comportamientos esperados (por ejemplo, procesos de negocio soportados o restricciones del cliente que el servicio eliminará).
- **Requisitos de gestión y operacionales:** Son requisitos no funcionales que establecen las condiciones y restricciones bajo las cuales el servicio debe operar. Incluyen necesidades de disponibilidad, desempeño, seguridad, facilidad de despliegue, mantenibilidad y otros aspectos de garantía (*warranty*) del servicio.
- **Requisitos de usabilidad:** Relacionados con la facilidad de uso y accesibilidad del servicio por parte de los usuarios finales. Abarcan consideraciones sobre la interfaz de usuario y la experiencia de usuario (“look and feel”), asegurando que el servicio sea intuitivo y eficiente de utilizar.

## 1.2. Técnicas de investigación de requisitos

Para obtener y refinar los requisitos, se emplea una combinación de técnicas que permiten descubrir información explícita e implícita de los usuarios y del negocio. No existe una única técnica suficiente; a menudo es necesario combinar varias para capturar diferentes perspectivas. Algunas técnicas comunes incluyen:

- **Entrevistas:** Consisten en reuniones uno a uno (o en grupos pequeños) con usuarios, clientes u otros interesados, siguiendo un guion de preguntas para identificar

necesidades, expectativas y cualquier requisito específico..

- **Talleres (workshops):** Sesiones grupales facilitadas donde múltiples *stakeholders* trabajan conjuntamente para identificar, discutir y priorizar requisitos. Los talleres fomentan la colaboración y pueden revelar información importante a través de la interacción entre participantes.
- **Observación directa:** Implica presenciar cómo los usuarios realizan sus tareas cotidianas para entender el contexto real de uso. Mediante la observación (y técnicas como el *shadowing*, donde el analista acompaña al usuario durante su jornada), se identifican requisitos que los usuarios pueden pasar por alto al explicarlos, ya que ciertas actividades se dan por supuestas.
- **Análisis de protocolos:** Consiste en pedir a los usuarios que realicen tareas típicas mientras piensan en voz alta, o en analizar registros detallados de cómo realizan sus procesos.
- **Análisis de escenarios:** Se desarrollan escenarios hipotéticos o casos de uso concretos que describen situaciones de negocio que el servicio debe soportar. Los usuarios revisan y discuten estos escenarios, lo cual no solo valida requisitos conocidos sino que también puede descubrir requisitos tácitos al considerar cómo reaccionaría el servicio en distintas situaciones.
- **Prototipado:** Se construyen prototipos o versiones simplificadas del servicio o de la aplicación (por ejemplo, modelos de pantalla o funcionalidad limitada) para mostrar a los usuarios.

### 1.3. Problemas comunes y soluciones en la ingeniería de requisitos

Durante el proceso de ingeniería de requisitos suelen surgir diversos problemas que, de no abordarse, pueden comprometer el resultado final:

- **Requisitos conflictivos o incompatibles:** Distintos *stakeholders* pueden tener necesidades u objetivos que chocan entre sí, llevando a requisitos mutuamente excluyentes. Por ejemplo, el área de seguridad podría requerir controles estrictos que dificultan la usabilidad requerida por los usuarios.
- **Requisitos omitidos o poco claros:** A veces ciertos requisitos no se identifican inicialmente (especialmente si son tácitos) o se describen de forma ambigua, dando lugar a interpretaciones erróneas. Si aspectos críticos se descubren tarde, pueden causar retrabajo o fallos en el servicio. Para mitigarlo, se debe involucrar a una variedad de usuarios desde el inicio, revisar los requisitos con ellos (validación iterativa) y mantener un glosario común para aclarar términos y evitar malentendidos.

- **Cambios y desbordamiento de alcance:** Los requisitos pueden evolucionar durante el proyecto (cambios de negocio, regulaciones nuevas, etc.), o bien el alcance puede ampliarse sin control (*scope creep*). Esto dificulta cerrar un conjunto estable de requisitos y puede impactar en costos y plazos. La solución incluye gestionar formalmente los cambios (control de cambios y trazabilidad), estableciendo un proceso para evaluar el impacto de cada nuevo requerimiento y obtener la aprobación adecuada antes de incorporarlo.
- **Prioridades indefinidas:** Si todos los requisitos se tratan como igualmente importantes, el equipo de diseño podría no saber en qué concentrarse en caso de limitaciones de tiempo o recursos. Es crucial asignar prioridades (por ejemplo, usando la técnica MoSCoW: *Must, Should, Could, Won't*) para distinguir requisitos imprescindibles de aquellos deseables.

## 1.4. Conocimiento tácito y explícito en los requisitos

Un desafío importante en la ingeniería de requisitos es capturar tanto el conocimiento explícito como el tácito de los usuarios y la organización:

**Conocimiento explícito** es aquel que los usuarios pueden articular fácilmente: reglas de negocio, pasos de un procedimiento formal, datos que manejan cotidianamente y que están documentados o claramente presentes en su mente. Este tipo de conocimiento suele aparecer en entrevistas y documentos existentes.

**Conocimiento tácito**, por el contrario, es el saber acumulado que los usuarios poseen pero que les resulta difícil expresar de forma directa. Incluye intuiciones, experiencias prácticas, atajos y detalles del trabajo diario que no están formalmente documentados. A menudo, los usuarios no mencionan estos aspectos porque los dan por supuestos o no son conscientes de ellos hasta que se enfrentan a situaciones específicas.

Las técnicas de investigación orientadas a escenarios y prototipos son especialmente útiles para revelar conocimiento tácito. Al simular situaciones reales o experimentar con una interfaz preliminar, los usuarios pueden recordar excepciones, condiciones especiales o necesidades adicionales que no surgieron en conversaciones abstractas. Asimismo, el *shadowing* y la observación directa del trabajo ayudan a que el analista perciba detalles no verbalizados por el usuario.

## 2. Outsourcing en la Ingeniería de Requisitos

En algunos casos, el desarrollo de servicios o sistemas de TI se subcontrata total o parcialmente a proveedores externos. La externalización de la ingeniería de requisitos implica decidir cómo se dividirá la responsabilidad de especificar los requisitos

entre la organización cliente y el proveedor. ITIL sugiere que, siempre que sea posible, se consideren soluciones empaquetadas estándar (*software comercial*) para satisfacer las necesidades, ya que pueden reducir tiempo y esfuerzo de desarrollo.

Existen distintos escenarios típicos para especificar requisitos en contratos de desarrollo externalizado:

- **Especificación de bajo nivel por el cliente:** En este enfoque, la organización cliente proporciona al proveedor una especificación detallada de todos los requisitos (particularmente de aquellos que impactan al usuario final). La frontera entre el cliente y el proveedor se traza después de la fase de especificación minuciosa; es decir, el proveedor se encarga del diseño técnico y la construcción con un objetivo muy claro y preciso.
- **Especificación de alto nivel:** En este caso, el cliente define solo los requisitos a alto nivel (objetivos de negocio, capacidades globales deseadas) y el proveedor asume la responsabilidad de detallar el diseño, construcción e implementación. La frontera cliente/proveedor se sitúa inmediatamente después de la definición de requisitos generales: el contrato con el proveedor cubre desde el diseño detallado hasta la entrega.

### 3. Gestión de Datos e Información

Los datos son uno de los activos críticos que una organización debe gestionar adecuadamente para poder desarrollar, entregar y dar soporte a los servicios de TI de forma eficaz. La gestión de datos e información abarca cómo la organización planifica, recolecta, crea, organiza, utiliza, controla, distribuye y finalmente dispone de sus datos, tanto en forma de registros estructurados (p.ej. bases de datos) como de información no estructurada (documentos, correos, etc.).

Es importante notar que la gestión de datos no se limita al dato bruto en sí, sino también al *metadato* asociado (es decir, “datos sobre los datos” como descripciones, definiciones, origen, contexto). Cuando los datos se enriquecen con contexto a través de metadatos, se convierten efectivamente en información útil para la toma de decisiones.

Para que los datos de la organización sean tratados verdaderamente como un activo, deben cumplir ciertos criterios: deben ser fácilmente accesibles para quienes los necesitan, compartidos apropiadamente (evitando silos), mantenidos con niveles de calidad adecuados (exactitud, confiabilidad, consistencia), protegidos según requerimientos legales (privacidad, confidencialidad, integridad) y gestionados de forma eficiente y efectiva en términos de costo y uso.

### **3.1. Ciclo de vida de los datos**

ITIL recomienda adoptar un enfoque de ciclo de vida para entender y gestionar el uso de los datos a lo largo de todos los procesos de negocio. Esto implica considerar desde la creación o captura inicial del dato, su almacenamiento y uso en las operaciones diarias, hasta su archivo o eliminación final.

Responder a estas preguntas ayuda a diseñar procesos y controles para cada etapa del ciclo de vida de la información. Por ejemplo, durante el diseño del servicio, la función de gestión de datos puede aconsejar sobre modelos de datos apropiados, normas de formato y técnicas para convertir modelos lógicos de datos en implementaciones físicas eficientes.

### **3.2. Clasificación y estándares de la información**

Una parte fundamental de la gobernanza de datos es la clasificación de la información. Clasificar los datos significa categorizarlos según criterios relevantes para la organización, como su nivel de sensibilidad, criticidad para el negocio o tipo de uso. Por ejemplo, se pueden etiquetar datos como públicos, confidenciales, secretos; o como datos maestros, transaccionales, de referencia, etc. La clasificación adecuada permite aplicar controles y tratamientos diferenciados: datos altamente confidenciales requerirán mayores medidas de seguridad, datos críticos para el negocio pueden tener réplicas y respaldos más frecuentes, mientras que datos poco usados podrían archivarse tras cierto tiempo.

Ligados a la clasificación están los estándares de datos. La organización debe definir políticas y estándares para la gestión de datos que cubran aspectos como formatos permitidos, convenciones de nombres, calidad mínima aceptable y cumplimiento normativo.

### **3.3. Propiedad de los datos**

Un principio clave en la gestión de datos es establecer claramente la propiedad o responsabilidad sobre los datos y sus definiciones. Cada conjunto de datos importante (por ejemplo, una base de datos de clientes, o un repositorio de documentos) debería tener un dueño asignado dentro de la organización.

## **Referencias**

Office of Government Commerce (OGC). (2011). *Itil v3: Service design*. London, UK: TSO (The Stationery Office).