

# CICLO DE VIDA DE UN SISTEMA

Un sistema informático puede ser definido como un sistema de información que basa la parte fundamental de su procesamiento en el empleo de la computación. Como cualquier sistema, es un conjunto de funciones interrelacionadas, hardware, software y de Recurso Humano.

Se trata de las fases por las que debe pasar un sistema de informática en cuanto al análisis y el diseño para mantener la estructura y el funcionamiento de otro sistema desarrollado, tanto web como empresarial.

## Análisis

**Propósito:** Capturar, entender y documentar las necesidades y expectativas del cliente o usuario final. El objetivo principal es construir una base sólida para todo el proyecto.

### Actividades:

- Reuniones con los stakeholders (clientes, usuarios y otros interesados).
- Identificación de objetivos, problemas actuales y necesidades del sistema.
- Elaboración de documentos como especificaciones funcionales, diagramas de casos de uso y listas de requisitos.
- Análisis de viabilidad para verificar si los objetivos son técnicamente posibles, económicamente viables y legalmente aceptables.

**Importancia:** Es la columna vertebral del proyecto. Si no se entienden correctamente los requerimientos desde el inicio, es probable que el sistema final no cumpla las expectativas, lo que puede resultar en costosos retrabajos y pérdida de tiempo.

## Diseño

**Propósito:** Traducir los requisitos en una arquitectura o modelo técnico que detalle cómo se desarrollará el sistema.

### Actividades:

- Diseñar la arquitectura del sistema (bases de datos, servidores, módulos, etc.).
- Crear prototipos y diagramas como diagramas de flujo, diagramas de clases y diagramas de secuencia.
- Elegir las tecnologías, lenguajes de programación y herramientas que se usarán en el desarrollo.
- Planificar interfaces de usuario y experiencia (UI/UX).

**Importancia:** Proporciona un plano claro para los desarrolladores, minimizando la ambigüedad y asegurando que el sistema sea escalable, eficiente y fácil de mantener.

## Desarrollo

**Propósito:** Construir el sistema a partir del diseño creado, es decir, escribir el código que dará vida al sistema.

**Actividades:**

- Programación de los módulos y funcionalidades.
- Pruebas unitarias para asegurar que cada módulo funcione correctamente.
- Integración de módulos para garantizar que trabajen juntos sin problemas.
- Gestión de versiones y control de calidad del código.

**Importancia:** Es la etapa donde el sistema realmente empieza a existir. Un desarrollo bien estructurado asegura que el producto final sea funcional, eficiente y libre de errores graves.

## Implementación

**Propósito:** Poner el sistema en funcionamiento en un entorno real y asegurarse de que funcione correctamente para los usuarios finales.

**Actividades:**

- Configuración de servidores y bases de datos en el entorno de producción.
- Capacitación de los usuarios finales y entrega de manuales o documentación.
- Realización de pruebas de aceptación para garantizar que el sistema cumple con los requisitos.
- Migración de datos desde sistemas antiguos, si aplica.

**Importancia:** El sistema comienza a interactuar con los usuarios finales. Una implementación bien ejecutada garantiza una transición suave y aumenta la confianza en el sistema.

## Mantenimiento

**Propósito:** Garantizar que el sistema siga siendo funcional, eficiente y relevante a lo largo del tiempo.

**Actividades:**

- Corrección de errores reportados por los usuarios (mantenimiento correctivo).
- Incorporación de nuevas funcionalidades o mejoras (mantenimiento evolutivo).
- Adaptación del sistema a nuevos entornos tecnológicos o legales (mantenimiento adaptativo).
- Supervisión del desempeño del sistema y optimización continua.

**Importancia:** Es una fase clave para prolongar la vida útil del sistema. El mantenimiento asegura que el sistema pueda evolucionar junto con las necesidades de los usuarios y los cambios en el entorno tecnológico.

Cada fase del ciclo de vida de un sistema está interconectada. Un buen análisis asegura un diseño sólido, un diseño sólido facilita un desarrollo efectivo, y una implementación exitosa sienta las bases para un mantenimiento sostenible. Por eso, ninguna fase debe tomarse a la ligera: cada una tiene un rol crítico en el éxito del proyecto.