Logotipo

Descripción generada automáticamenteIcono

Descripción generada automáticamente**Centro Universitario de Ciencias Exactas e**

**Ingenierías**

***Universidad de Guadalajara***

Actividad 1.1: Introducción a la Ingeniería de Software

*Ingeniería de Software*



**Alumno:** Samuel David Pérez Brambila

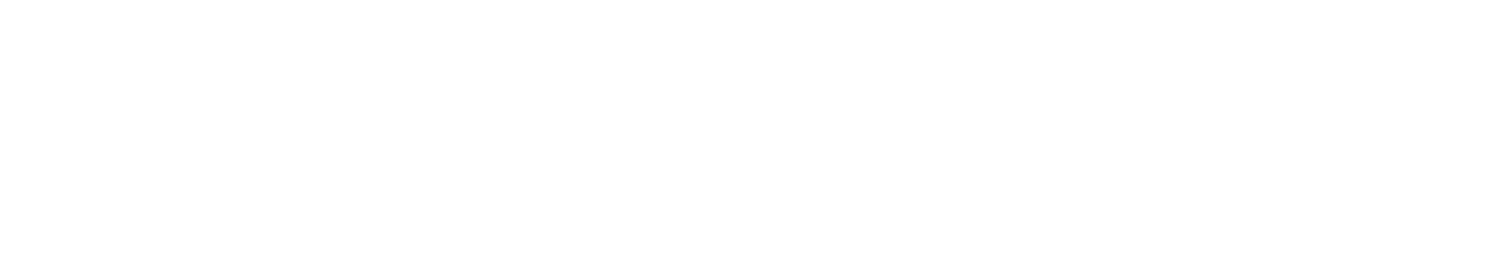
**Código:** 222966286

**Profesor:** Mauricio Rodolfo Arreola González

**Sección:** D09

**Fecha de Entrega:**

24 de Enero de 2025



**Índice**

[Definición de software 1](#_Toc188105306)

[Definición de ingeniería de software 1](#_Toc188105307)

[Tipos de software y para qué sirve cada uno 1](#_Toc188105308)

[Software de sistema 1](#_Toc188105309)

[Software de programación 2](#_Toc188105310)

[Software de aplicación 2](#_Toc188105311)

[Malware o software malicioso 3](#_Toc188105312)

[Definición de proceso del software 4](#_Toc188105313)

[Fases en el desarrollo de software y acciones que se llevan a cabo en cada una 4](#_Toc188105314)

[Planificación 4](#_Toc188105315)

[Definición de requisitos 4](#_Toc188105316)

[Diseño y prototipo 5](#_Toc188105317)

[Desarrollo del software 5](#_Toc188105318)

[Pruebas 5](#_Toc188105319)

[Lanzamiento 6](#_Toc188105320)

[Operaciones y mantenimiento 6](#_Toc188105321)

[Documentación 6](#_Toc188105322)

[Bibliografía 7](#_Toc188105323)

# **Definición de software**

El software es un término que hace referencia a un programa o conjunto de programas diseñados para realizar tareas específicas en un equipo informático, sea una computadora, un smartphone, entre otros. En otras palabras, el software engloba los programas, las reglas y las instrucciones que permiten que un equipo informático ejecute diversas funciones.

El software funciona mediante un lenguaje de programación conocido como código binario, representado por ceros y unos. Este código permite a los programadores instruir al dispositivo sobre qué acciones realizar, cómo y cuándo ejecutarlas.

# **Definición de ingeniería de software**

La ingeniería de software es una disciplina que implica el uso de estructuras, herramientas y técnicas para construir programas informáticos. Así mismo, incluye el análisis previo de la situación, la redacción del proyecto, la creación del software y las pruebas necesarias para garantizar el correcto funcionamiento del software antes de poner el sistema en funcionamiento.

Esta ingeniería aborda todas las fases del ciclo de vida de desarrollo de cualquier tipo de sistema de información y es aplicable a una amplia gama de ámbitos de la informática, como el diseño de compiladores, sistemas operativos y tecnologías de Intranet/Internet.

# **Tipos de software y para qué sirve cada uno**

Si realizamos una clasificación por funcionalidad, podemos distinguir entre cuatro tipos de software diferentes, los cuales son:

* **Software de sistema:** Es la base sobre la que se asienta el funcionamiento de cualquier equipo informático. Sus funciones principales son gestionar y coordinar los recursos del hardware para que todos los componentes y programas sean compatibles unos con otros. Este tipo de software es imprescindible para que los dispositivos puedan ejecutar las tareas que se les piden. Los tipos más comunes de software de sistema incluyen:
  + Cargadores de programa: Son programas pequeños que cargan el sistema operativo en la memoria del equipo cuando se enciende, para que el hardware se prepare para ejecutar tareas.
  + Sistemas operativos: Son la parte más importante del software de sistema, ya que gestionan todos los recursos del hardware. Permite que las aplicaciones se ejecuten y que haya una interfaz para que los usuarios interactúen con el equipo.
  + Controladores: Estos programas permiten que el SO se comunique y controle dispositivos externos para que puedan funcionar.
  + Herramientas de diagnóstico: Son programados diseñados para monitorear y analizar el rendimiento del hardware y del sistema operativo. Identifican problemas y sugieren soluciones para que el equipo se mantenga en buen estado.
  + Servidores: Son programas que gestionan los recursos de red para que varios dispositivos compartan estos, así como información. Pueden ser locales o basados en la nube.
* **Software de programación:** El software de programación es la herramienta que utilizan los desarrolladores para crear el resto de programas que usamos día con día. Estos tipos de software crean el entorno que se necesita para escribir, probar y perfeccionar el código fuente que luego se convertirá en aplicaciones o sistemas operativos. Sin este, no sería posible crear nuevos programas, ya que se necesita una traducción de las ideas en instrucciones que sean comprensibles para los ordenadores.

Algunos ejemplos son PyCharm, Eclipse y Lazarus. Cada uno cuenta con un conjunto específico de herramientas y características que están adaptadas a diferentes lenguajes de programación y necesidades de desarrollo.

* **Software de aplicación:** El software de aplicación son los programas que se diseñan para que los usuarios puedan realizar tareas o funciones concretas. Es decir, es el producto final con el que interactuamos y nos permite, por ejemplo, redactar documentos, gestionar datos, comunicarnos o disfrutar de entretenimiento. Estos tipos de software abarcan distintas herramientas que se verán más adelante, pero todas ellas buscan satisfacer necesidades específicas del usuario, es por ello que en su desarrollo se tiene muy en cuenta la experiencia del usuario y la funcionalidad.

Los tipos más comunes son los siguientes:

* + Aplicaciones de ofimática: Facilitan las tareas de oficina como crear documentos, realizar presentaciones u hojas de cálculo. Algunos ejemplos son Microsoft Office, Google Workspace y LibreOffice. Actualmente son fundamentales para la productividad en entornos de trabajo y espacios educativos.
  + Bases de datos: Las bases de datos son programas que se especializan en crear, gestionar y analizar grandes volúmenes de datos. Por ejemplo, MySQL, PostgreSQL, Microsoft Access; que son aplicaciones que se usan para organizar información y acceder a ella.
  + Videojuegos: Son aplicaciones diseñadas para el entretenimiento interactivo, pero también permiten los avances tecnológicos en gráficos y diseño de interacción.
  + Software empresarial: Se trata de herramientas creadas para gestionar procesos y operaciones en una empresa. Por ejemplo, SAP o Salesforce.
  + Software educativo: Facilitan el aprendizaje y la enseñanza. Pueden ser plataformas de e-learning como Moodle o aplicaciones interactivas como Duolingo.
  + Software de gestión: Se usan para administrar y organizar tareas, proyectos o recursos. Por ejemplo, QuickBooks para la contabilidad, Trello para la gestión de proyectos y Jira para el seguimiento de tareas en el desarrollo de software.
* **Malware o software malicioso:** El malware, también llamado software malicioso, es un tipo de software diseñado con la función concreta de infiltrarse, dañar o acceder a los sistemas informáticos sin autorización. Su objetivo es robar datos, espiar actividades, corromper archivos o deshabilitar sistemas. Forma parte de actividades de ciberdelicuencia y se suele propagar por correos electrónicos, descargas no seguras o las vulnerabilidades en el software. Podemos encontrar varios tipos de malware como los siguientes:
  + Troyanos: Son programas que se disfrazan de un software legítimo para hacer creer a los usuarios que son el original y los instalen en sus sistemas. Cuando se han instalado, crean puertas traseras en el sistema, para que las personas que lo han creado puedan acceder en remoto a la información sensible o instalen otros tipos del malware.
  + Spyware: Se instala en el sistema sin el conocimiento del usuario y recopila su información personal. Por ejemplo, contraseñas, historial de navegación o datos financieros.
  + Ransomware: Cifra los archivos de la víctima y exige un pago, conocido como ransom, a cambio de la clave de descifrado.
  + Gusanos: Son programas maliciosos que se replican a sí mismos y se propagan de un equipo a otro sin que haya intervención humana.
  + Adware: Aunque no siempre es malicioso, es un software que muestra anuncios no deseados en el sistema del usuario. También suele recopilar información sobre el usuario sin su consentimiento, por lo que vulnera la privacidad.
  + Rootkits: Se utiliza para ocultar la presencia de otros programas maliciosos en un sistema. Modifican el sistema operativo para que el software de seguridad no los detecte.

# **Definición de proceso del software**

Un proceso de software es una serie de actividades relacionadas que conduce a la elaboración de un producto de software. Estas actividades pueden incluir el desarrollo de software desde cero en un lenguaje de programación estándar como Java o C.

Existen diferentes procesos de software, pero todos deben incluir 4 actividades que son fundamentales para la ingeniería de software:

1. Especificación del software: Tienen que definirse tanto la funcionalidad del software como las restricciones de su operación.
2. Diseño e implementación del software: Debe desarrollarse el software para cumplir con las especificaciones.
3. Validación del software: Hay que validad el software para asegurarse de que cumple lo que el cliente quiere.
4. Evolución del software: El software tiene que evolucionar para satisfacer las necesidades cambiantes del cliente.

Tales actividades forman parte de todos los procesos de software. Por supuesto, en la práctica éstas son actividades complejas en sí mismas e incluyen en subactividades tales como la validación de requerimientos, el diseño arquitectónico, la prueba de unidad, etc. También existen actividades de soporte al proceso, como la documentación y el manejo de la configuración del software.

# **Fases en el desarrollo de software y acciones que se llevan a cabo en cada una**

El ciclo de desarrollo de software es un proceso estructurado que se divide en varias etapas, cada una esencial para asegurar la creación de un producto de alta calidad. Las etapas o fases en el ciclo de desarrollo de software son:

* **Planificación:** La planificación es la primera etapa de todo proyecto y es fundamental para su éxito. Aquí se definen los objetivos, se asignan recursos, se establece un cronograma y se identifican los posibles riesgos. Esta fase garantiza que todos los involucrados tengan una comprensión clara de lo que se va a desarrollar y cómo se logrará.
* **Definición de requisitos:** En esta etapa, se recogen y documentan las necesidades y expectativas del cliente. Para ello, se llevan a cabo entrevistas, encuestas y análisis para definir qué debe hacer el programa, cuáles son las características que se incluirán y los posibles obstáculos que podrían presentarse. El objetivo es crear una base sólida que guíe todo el desarrollo del proyecto. También deben incluirse los recursos disponibles y necesarios para integrarse.
* **Diseño y prototipo:** Una vez definidos los requisitos, se pasa a la fase de diseño. Aquí se crea la arquitectura del software y se desarrollan prototipos que representan la funcionalidad y la interfaz de usuario. Estos prototipos permiten visualizar el producto final y realizar ajustes antes de la codificación.
* **Desarrollo del software:** En esta etapa, los desarrolladores comienzan a escribir el código según las especificaciones del diseño. En esta fase se utilizan herramientas de desarrollo especializado para asegurar un proceso de codificación eficiente y organizado con metas establecidas y de esta forma los desarrolladores comprendan las expectativas del cliente. Cada destacar que, en algunos casos, la etapa de desarrollo puede fusionar con la etapa de prueba, la cual se realiza para garantizar que no haya errores críticos.
* **Pruebas:** El software desarrollado se somete a diversas pruebas para identificar y corregir errores. Se realizan pruebas unitarias, de integración, de sistema y de aceptación para asegurar que el software funcione correctamente y cumpla con los requisitos establecidos. Los tipos de prueba a realizar son:

* + Pruebas de rendimiento: Evalúan la velocidad, la capacidad de respuesta y la estabilidad del software bajo diversas condiciones de carga. Estas pruebas aseguran que el software pueda manejar grandes volúmenes de datos y usuarios simultáneos sin degradar su rendimiento.
  + Pruebas funcionales: Verifican que todas las funcionalidades del software operen conforme a los requisitos especificados. Estas pruebas se centran en comprobar que cada función del software produce los resultados esperados.
  + Pruebas de seguridad: Analizan la resistencia del software a posibles ataques y vulnerabilidades. Estas pruebas garantizan que los datos del usuario estén protegidos y que el sistema sea seguro frente a amenazas externas.
  + Pruebas unitarias: Se realizan en las unidades individuales de código para verificar su correcto funcionamiento. Estas pruebas son generalmente realizadas por los desarrolladores y ayudan a detectar errores en etapas tempranas del desarrollo.
  + Pruebas de usabilidad: Evalúan la facilidad de uso del software desde la perspectiva del usuario final. Estas pruebas aseguran que la interfaz sea intuitiva y que los usuarios puedan interactuar con el sistema de manera eficiente y efectiva.
  + Pruebas de aceptación: Validan que el software cumple con los requisitos y expectativas del cliente. Estas pruebas son realizadas por el cliente o un equipo de prueba externo y son cruciales para la aprobación final del producto antes de su lanzamiento.
* **Lanzamiento:** Una vez que el software ha superado todas las pruebas, se prepara para su lanzamiento. Esta etapa incluye la implementación del software en el entorno de producción, la capacitación de los usuarios y la realización de cualquier migración de datos necesaria.
* **Operaciones y mantenimiento:** La fase de operaciones y mantenimiento es continua y asegura que el software siga funcionando correctamente y se mantenga actualizado. Incluye la corrección de errores, la implementación de mejoras y el soporte al usuario.
* **Documentación:** Se debe dejar registro documentado de todo el proceso y cada una de las etapas del proyecto, considerando las modelaciones, los diagramas, las pruebas, el objetivo de las eventuales correcciones, la usabilidad y las posibles adecuaciones al sistema. También, deben realizarse el manual de usuario y el manual técnico.

# **Bibliografía**

* Carmona, B. (2025, 5 de enero). *¿Qué es un software? Definición, para qué sirve, tipos y beneficios*. Taclia. <https://www.taclia.com/blog/que-es-un-software>
* Costanzo, M. (s.f.). *¿Qué es un proceso de software?*. Mwebs. Consultado el 18 de Enero de 2025. <https://mwebs.com.uy/blog/qu%C3%A9-es-un-proceso-de-software/15>
* INESDI. (2024, 4 de septiembre). *Qué tipos de software existen y cómo funcionan*. INESDI Business Techschool. <https://www.inesdi.com/blog/tipos-de-software/>
* Ramírez, O. (2024, 29 de julio). *7 Fases de ciclo de desarrollo de software*. Bambú Mobile. <https://bambu-mobile.com/ciclo-de-desarrollo-de-software/>
* UNIR. (2021, 6 de abril). *Ingeniería de software: qué es, objetivos y funciones del ingeniero*. UNIR México. <https://mexico.unir.net/noticias/ingenieria/ingenieria-de-software-que-es-objetivos/>