

**U**

**P**

**T**

**UNIVERSIDAD POLITÉCNICA DE TULANCINGO**

**REINGENIERÍA DE SOFTWARE**

por

**Erick Adier Ortiz Cabrera**

**Ingeniería en Sistemas Computacionales**

Asignatura:

**Gestión de Desarrollo de Software**

Nombre del Catedrático:

**Carlos Alberto Marquez Sosa**

Tulancingo de Bravo, Hidalgo

Enero - Abril 2025



***ÍNDICE***

[***¿Qué es la reingeniería de software?*** 3](#_Toc191760974)

[***Objetivos de la Reingeniería de Software*** 3](#_Toc191760975)

[***Proceso de Reingeniería de Software*** 3](#_Toc191760976)

[***Herramientas y Técnicas Utilizadas*** 4](#_Toc191760977)

[***Beneficios y Desafíos*** 4](#_Toc191760978)

[***REFERENCIAS*** 5](#_Toc191760979)

# ***¿Qué es la reingeniería de software?***

La reingeniería de software es un enfoque estructurado que consiste en analizar, rediseñar y reimplementar sistemas de software legacy (heredados) para mejorar su calidad, rendimiento y capacidad de mantenimiento. No se trata simplemente de corregir errores o agregar nuevas funcionalidades, sino de transformar el sistema desde su base, manteniendo su esencia, pero mejorando su arquitectura y código.

Este proceso es especialmente útil cuando un sistema se ha vuelto obsoleto, difícil de mantener o no cumple con los requisitos actuales del negocio. La reingeniería no implica crear un sistema desde cero, sino más bien reconstruirlo de manera eficiente.

# ***Objetivos de la Reingeniería de Software***

Los principales objetivos de la reingeniería de software son:

1. **Mejorar la mantenibilidad**: Facilitar la corrección de errores y la incorporación de nuevas funcionalidades.
2. **Optimizar el rendimiento**: Asegurar que el sistema funcione de manera más eficiente.
3. **Reducir costos**: Minimizar los gastos asociados al mantenimiento y operación del software.
4. **Adaptar a nuevas tecnologías**: Integrar herramientas y frameworks modernos que permitan al sistema seguir siendo relevante.
5. **Documentar el sistema**: Crear o actualizar la documentación para que sea comprensible y útil para los desarrolladores.

# ***Proceso de Reingeniería de Software***

La reingeniería de software sigue un proceso estructurado que generalmente incluye las siguientes etapas:

1. **Análisis del sistema existente**:
   * Evaluar el estado actual del software, identificando sus fortalezas, debilidades y áreas de mejora.
   * Revisar el código fuente, la documentación y la arquitectura del sistema.
2. **Reestructuración del código**:
   * Refactorizar el código para mejorar su legibilidad y eficiencia sin alterar su comportamiento externo.
   * Eliminar código redundante o innecesario.
3. **Rediseño de la arquitectura**:
   * Modificar la estructura del sistema para hacerla más modular y escalable.
   * Adoptar patrones de diseño y buenas prácticas de desarrollo.
4. **Reimplementación**:
   * Reescribir partes del sistema utilizando tecnologías modernas.
   * Integrar nuevas funcionalidades o mejorar las existentes.
5. **Pruebas y validación**:
   * Asegurar que el sistema reingenierizado funcione correctamente y cumpla con los requisitos establecidos.
   * Realizar pruebas unitarias, de integración y de rendimiento.
6. **Documentación**:
   * Actualizar la documentación técnica y de usuario para reflejar los cambios realizados.

# ***Herramientas y Técnicas Utilizadas***

Existen diversas herramientas y técnicas que facilitan el proceso de reingeniería de software. Algunas de las más comunes incluyen:

* **Herramientas de análisis estático**: Como SonarQube, que ayudan a identificar problemas en el código.
* **Herramientas de refactorización**: Integradas en IDEs como IntelliJ IDEA o Visual Studio.
* **Frameworks de pruebas**: Como JUnit o Selenium, para validar el funcionamiento del sistema.
* **Técnicas de minería de datos**: Para extraer información útil de sistemas legacy.

# ***Beneficios y Desafíos***

Lla reingeniería de software ofrece numerosos beneficios, como la extensión de la vida útil de los sistemas, la reducción de costos a largo plazo y la mejora de la satisfacción del usuario. Sin embargo, también presenta desafíos, como la complejidad del proceso, la necesidad de recursos especializados y el riesgo de introducir nuevos errores durante la reimplementación.

# ***REFERENCIAS***

Imagar. (2023, 5 septiembre). *El proceso de reingeniería de software*. Imagar Solutions Company. <https://www.imagar.com/blog-desarrollo-web/el-proceso-de-reingenieria-de-software/>

*Reingeniería del software*. (2011, 26 septiembre). <https://juanantoniomb.wordpress.com/reingenieria-del-software-2/>